

PATENT ABSTRACTS OF JAPAN

(11) Publication number : 2000-307641
 (43) Date of publication of application : 02. 11. 2000

(51) Int. Cl.

H04L 12/56
 G06F 17/30

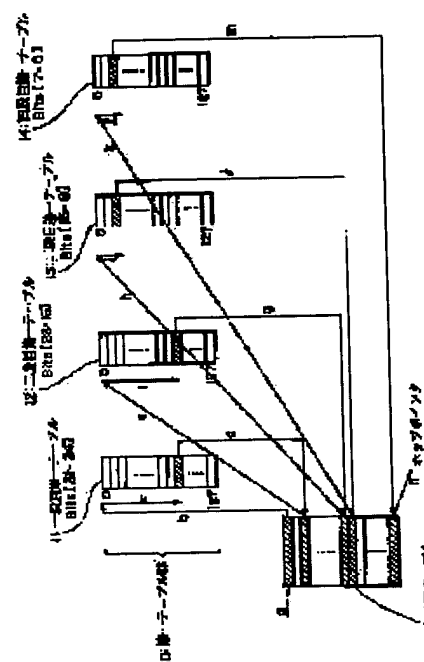
(21) Application number : 11-110041 (71) Applicant : NEC CORP
 (22) Date of filing : 16. 04. 1999 (72) Inventor : ARAMAKI TOSHIYA
 OKAMOTO TSUGIO

(54) METHOD AND DEVICE FOR TRANSFER DESTINATION RETRIEVAL, AND RETRIEVAL TABLE RECORDING MEDIUM AND RETRIEVAL PROGRAM RECORDING MEDIUM

(57) Abstract:

PROBLEM TO BE SOLVED: To actualize transfer destination retrieval technology which can save memory capacity and makes it easy to change a registered address.

SOLUTION: Divided bit streams obtained by dividing an IP address by 8 bits into four parts are made to correspond to entry addresses of 1st tables 11 to 14 of 1st to 4th stages from the high order, only a 2nd table pointer indicating the entry of the 2nd table 20 is set in the respective 1st tables 11 to 14, and a 1st table pointer and a hop pointer indicating the 1st table of a next stage are set in the 2nd table 20, and retrieval is carried out by using the entry addresses of the 1st tables as offsets as they are and the 2nd and 1st tables are accessed alternately according to the 1st and 2nd table pointers to retrieve a hop pointer indicating a next transfer destination.



LEGAL STATUS

[Date of request for examination] 24. 03. 2000
 [Date of sending the examiner's decision of rejection]
 [Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]
 [Date of final disposal for application]
 [Patent number] 3250544

[Date of registration] 16.11.2001
[Number of appeal against examiner's
decision of rejection]
[Date of requesting appeal against
examiner's decision of rejection]
[Date of extinction of right]

Copyright (C); 1998, 2003 Japan Patent Office

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号
特開2000-307641
(P2000-307641A)

(43)公開日 平成12年11月2日(2000.11.2)

(51)Int.Cl.⁷

識別記号

F I

テ-マ-ト* (参考)

H 0 4 L 12/56

H 0 4 L 11/20

1 0 2 D 5 B 0 7 5

G 0 6 F 17/30

G 0 6 F 15/40

3 7 0 Z 5 K 0 3 0

15/413

3 1 0 A

審査請求 有 請求項の数11 O L (全 25 頁)

(21)出願番号

特願平11-110041

(22)出願日

平成11年4月16日(1999.4.16)

(71)出願人 000004237

日本電気株式会社

東京都港区芝五丁目7番1号

(72)発明者 荒巻 利也

東京都港区芝五丁目7番1号 日本電気株式会社内

(72)発明者 岡本 継男

東京都港区芝五丁目7番1号 日本電気株式会社内

(74)代理人 100086759

弁理士 渡辺 喜平

Fターム(参考) 5B075 ND04 NK43 NK54 PQ05

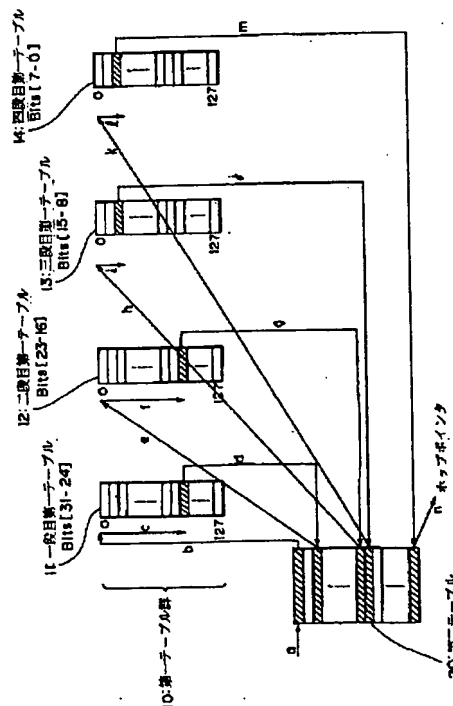
5K030 HA08 HB11 KA05 LB05

(54)【発明の名称】 転送先検索方法、転送先検索装置、検索テーブル記録媒体及び検索プログラム記録媒体

(57)【要約】

【課題】 メモリ容量が節約でき、かつ、登録アドレスの変更が容易な転送先検索技術の提供。

【解決手段】 I Pアドレスを8ビットずつ四つに分割して選られた分割ビット列を、上位から一段目～四段目第一テーブル11～14のエントリアドレスに対応させ、各第一テーブルには、第二テーブル20のエントリを示す第二テーブルポイントのみを設定し、第二テーブルには、次段の第一テーブルを示す第一テーブルポイント及びホップポイントを設定しておき、各第一テーブルのエントリアドレスをそのままオフセットとして検索し、第一及び第二テーブルポイントにしたがって、第二及び第一テーブルに交互にアクセスして、次の転送先を示すホップポイントを検索する。



【特許請求の範囲】

【請求項 1】 転送されてきたデータ信号の次の転送先を決定するために、当該データ信号の送信先アドレスを検索キーとして、検索テーブルから、当該データ信号の転送先の登録アドレスを示すホップポインタを検索する転送先検索方法において、
前記検索テーブルを、第一テーブル群と第二テーブルとにより構成しておき、
前記第一テーブル群は、前記送信先アドレスに割当てられたビット数分のビット列を上位から一定ビットずつ複
数段に分割して得られた各段の分割ビット列にそれぞれ
対応する複数の第一テーブルからなり、
各前記第一テーブルは、当該第一テーブルの対応する段の分割ビット列で表される各分割ビット列値を、当該第一テーブルの各エントリのオフセットアドレスとしてそれぞれ有するとともに、
前記オフセットアドレスが、登録アドレスのビット列値のうち当該段に対応する分割ビット列値と一致する場合に、当該第一テーブルは、当該オフセットアドレスのエントリ内容として、次にアクセスする前記第二テーブル
のエントリを示す第二テーブルポインタを有し、かつ、
前記登録アドレスの有効ビット列の最下位ビットが前記分割ビット列の途中までしかない場合に、当該第一テーブルは、当該有効ビット列のうちの当該分割ビット列に相当する有効ビット列部分を共通して含む各オフセット
アドレスのエントリ内容として、互いに共通の前記第二
テーブルポインタを有し、
前記第二テーブルは、エントリ内容として、検索継続又は終了を指示する継続／終了フラグ用フィールド、ホップ
ポインタ用フィールド、及び、次にアクセスする第一
テーブルを示す第一テーブルポインタ用フィールドを有し、
前記ホップポインタを検索するにあたり、
前記送信先アドレスのビット列値を上位から前記一定ビ
ットずつ複数段に分割して得られる各段の分割ビット列
値のうち最上段の分割ビット列値をキーとして、前記第
一テーブル群のうち最上段の第一テーブルを検索し、当
該分割ビット列値と一致するオフセットアドレスのエ
ントリから前記第二テーブルポインタを読み出し、
前記第二テーブルポインタの示す前記第二テーブルのエ
ントリの継続／終了フラグが検索継続を指示している場
合には、前記第一テーブルポインタの示す次段の第一テ
ーブルを、前記送信先アドレスの当該段の分割ビット列
値をキーとして検索し、当該分割ビット列値と一致する
オフセットアドレスのエントリから前記第二テーブルポ
インタを繰返し読み出し、
前記第二テーブルポインタの示す第二テーブルのエ
ントリの継続／終了フラグが検索終了を指示している場
合には、当該エントリのホップポインタを読み出して検索を
終了することを特徴とする転送先検索方法。

【請求項 2】 前記第二テーブルの各エントリは、ホップ
ポインタ更新の有無を指示する更新フラグ用フィール
ドを有し、
前記ホップポインタを検索するにあたり、
初期値の前記ホップポインタを設定して、内部変数とし
て保持し、
前記第二テーブルポインタの示す前記第二テーブルのエ
ントリの前記更新フラグが更新を指示している場合に、
前記内部変数として保持されてホップポインタを、当該
エントリから読み出したホップポインタに更新すること
を特徴とする請求項 1 記載の転送先検索方法。

【請求項 3】 前記第一テーブルのエントリは、前記ホ
ップポインタを更新する必要がある場合にのみ、前記第
二テーブルポインタを有し、更新不要の場合には、前記
第二テーブルポインタの代わりに、前記登録アドレスの
ビット列値に従って次にアクセスする次段の第一テー
ブルを示す第一テーブルポインタを有し、
前記ホップポインタを検索するにあたり、前記第一テー
ブルのエントリに第一テーブルポインタが設定されてい
る場合には、前記第二テーブルを経由せずに、前記テー
ブルポインタの示す第一テーブルへ直接アクセスするこ
とを特徴とする請求項 1 又は 2 記載の転送先検索方法。

【請求項 4】 前記第一テーブルを、0 ビット用第一テ
ーブルと 1 ビット用第一テーブルとに分けて構成してお
き、

前記 0 ビット用第一テーブルは、前記第一テーブルの対
応する前記分割ビット列の先頭ビットの値が「0」であ
る場合の、当該分割ビット列から当該先頭ビットを除い
た残存分割ビット列に対応し、

前記 1 ビット用第一テーブルは、前記第一テーブルの対
応する前記分割ビット列の先頭ビットの値が「1」であ
る場合の、当該分割ビット列から当該先頭ビットを除い
た残存部分ビット列に対応し、

前記第二テーブルは、各エントリに、前記 0 ビット用第
一テーブルを示す 0 ビット用第一テーブルポインタ用フ
ィールドと、前記 1 ビット用第一テーブルを示す 1 ビ
ット用第一テーブルポインタ用フィールドとを有し、
前記ホップポインタを検索するにあたり、

分割ビット列の先頭ビット値が「0」であるときに、前
記 0 ビット用第一テーブルポインタの示す前記 0 ビット
用第一テーブルにアクセスし、
分割ビット列の先頭ビット値が「1」であるときに、前
記 1 ビット用第一テーブルポインタの示す前記 0 ビット
用第一テーブルにアクセスすることを特徴とする請求項
1、2 又は 3 記載の転送先検索方法。

【請求項 5】 前記第一テーブルが、前記登録アドレス
に対応するエントリを一つだけ有する場合に、当該第一
テーブルを当該エントリ一つ分だけのメモリ領域の記憶
容量を有する単独第一テーブルとし、

前記第二テーブルの各前記エントリは、当該エントリの

前記第一テーブルポインタの示す前記第一テーブルの種類が前記単独第一テーブルであるか否かを示すテーブル種類フラグ用フィールドを有し、

前記ホップポインタを検索するにあたり、

前記第二テーブルポインタの示す前記第二テーブルのエントリの前記テーブル種類フラグが、当該エントリの一テーブルポインタが示す第一テーブルが当該単独第一テーブルであることを示している場合に、当該単独第一テーブルに対応する登録アドレスの分割ビット列値と、前記送信先アドレスの有効ビット列のうち当該分割ビ

ット列に相当する有効ビット列部分とを照合し、当該分割ビット列値と当該有効ビット列部分とが不一致の場合に、検索を終了し、

当該分割ビット列値と当該有効ビット列部分とが一致した場合に、当該エントリの第二テーブルポインタの示す第二テーブルのエントリへアクセスすることを特徴とする請求項 1～4 のいずれかに記載の転送先検索方法。

【請求項 6】 前記単独第一テーブルを省略し、前記第二テーブルにおいて、当該単独第一テーブルを示す第一テーブルポインタの代わりに、前記単独第一テーブルのエントリに設定されるべき第二テーブルポインタを設定することを特徴とする請求項 5 記載の転送先検索方法。

【請求項 7】 前記第二テーブルの先頭エントリに、最上段の前記第一テーブルの先頭位置アドレスを示す先頭ポインタを格納しておく、

前記ホップポインタの検索を開始するにあたり、

前記第二テーブルの先頭エントリにアクセスし、当該先頭エントリの第一ポインタの示す第一テーブルにアクセスすることを特徴とする請求項 1～6 のいずれかに記載の転送先検索方法。

【請求項 8】 前記検索テーブルにおいて前記登録アドレスを変更するにあたり、

前記第二テーブルのエントリにおいて、前記第一テーブルポインタ及び前記ホップポインタを変更し、

前記第一テーブルのエントリにおいて、前記第二テーブルポインタを変更することを特徴とする請求項 1～7 のいずれかに記載の転送先検索方法。

【請求項 9】 ルータに転送されてきたデータ信号の次の転送先を決定するために、当該データ信号の送信先アドレスを検索キーとして、検索テーブルから、当該データ信号の転送先の登録アドレスを示すホップポインタを検索する転送先検索装置において、

前記検索テーブルを構成する第一テーブル群を格納した第一記憶部と、

前記検索テーブルを構成する第二テーブルを格納した第二記憶部と、

前記第一テーブル群へのアクセスを制御する第一制御部と、

前記第二テーブルへのアクセスを制御する第二制御部と

を備え、

前記第一テーブル群は、前記送信先アドレスに割当てられたビット数分のビット列を上位から一定ビットずつ複数段に分割して得られた各段の分割ビット列にそれぞれ対応する複数の第一テーブルからなり、

各前記第一テーブルは、当該第一テーブルの対応する段の分割ビット列で表される各分割ビット列値を、当該第一テーブルの各エントリのオフセットアドレスとしてそれぞれ有するとともに、

前記オフセットアドレスが、登録アドレスのビット列値のうち当該段に対応する分割ビット列値と一致する場合には、当該第一テーブルは、当該オフセットアドレスのエントリ内容として、次にアクセスする前記第二テーブルのエントリを示す第二テーブルポインタを有し、かつ、前記登録アドレスの有効ビット列の最下位ビットが前記分割ビット列の途中までしかない場合に、当該第一テーブルは、当該有効ビット列のうちの当該分割ビット列に相当する有効ビット列部分を共通して含む各オフセットアドレスのエントリ内容として、互いに共通の前記第二テーブルポインタを有し、

前記第二テーブルは、エントリ内容として、検索継続又は終了を指示する継続／終了フラグ用フィールド、前記ホップポインタ用フィールド、及び、次にアクセスする前記第一テーブルを示す第一テーブルポインタ用フィールドを有し、

前記第一制御部は、前記送信先アドレスを上位から前記一定ビットずつ複数段に分割して得られる各段の分割ビット列値のうち最上段の分割ビット列値をキーとして、前記第一テーブル群のうち最上段の前記第一テーブルを検索し、当該分割ビット列値と一致する前記オフセット

アドレスのエントリから前記第二テーブルポインタを読み出し、かつ、当該第二テーブルポインタを前記第二制御部へ転送し、

前記第二制御部は、前記第二テーブルポインタの示す前記第二テーブルのエントリの前記継続／終了フラグが検索継続を指示している場合に、前記第一テーブルポインタを読み出し、かつ、当該第一テーブルポインタを前記第一制御部へ転送し、

前記第一制御部は、前記第一テーブルポインタの示す次段の前記第一テーブルを、前記送信先アドレスの当該段の分割ビット列値をキーとして検索し、当該分割ビット列値と一致する前記オフセットアドレスのエントリから前記第二テーブルポインタを繰返し読み出し、かつ、当該第二テーブルポインタを前記第二制御部へ転送し、

前記第二制御部は、前記第二テーブルポインタの示す前記第二テーブルのエントリの前記継続／終了フラグが検索終了を指示している場合には、当該エントリの前記ホップポインタを読み出し、当該ホップポインタを次の転送先として出力することを特徴とする転送先検索装置。

【請求項 10】 転送されてきたデータ信号の次の転送

先を決定するために、当該データ信号の送信先アドレスを検索キーとして、当該データ信号の転送先の登録アドレスを示すホップポイントを検索するための検索テーブルの記録された、コンピュータ読み取り可能な記録媒体であって、

前記検索テーブルは、第一テーブル群と第二テーブルとにより構成され、

前記第一テーブル群は、前記送信先アドレスに割当てられたビット数分のビット列を上位から一定ビットずつ複数段に分割して得られた各段の分割ビット列にそれぞれ

対応する複数の第一テーブルからなり、
各前記第一テーブルは、当該第一テーブルの対応する段の分割ビット列で表される各分割ビット列値を、当該第一テーブルの各エントリのオフセットアドレスとしてそれぞれ有するとともに、

前記オフセットアドレスが、登録アドレスのビット列値のうち当該段に対応する分割ビット列値と一致する場合に、当該第一テーブルは、当該オフセットアドレスのエントリ内容として、次にアクセスする前記第二テーブルのエントリを示す第二テーブルポインタを有し、かつ、前記登録アドレスの有効ビット列の最下位ビットが前記分割ビット列の途中までしかない場合に、当該第一テーブルは、当該有効ビット列のうちの当該分割ビット列に相当する有効ビット列部分を共通して含むに各オフセットアドレスのエントリ内容として、互いに共通の前記第二テーブルポインタを有し、

前記第二テーブルは、エントリ内容として、検索継続又は終了を指示する継続／終了フラグ用フィールド、前記ホップポイント用フィールド、及び、次にアクセスする前記第一テーブルを示す第一テーブルポインタ用フィールドを有する検索テーブルが記録された記録媒体。

【請求項 11】 転送されてきたデータ信号の次の転送先を決定するために、当該データ信号の送信先アドレスを検索キーとして、検索テーブルから、当該データ信号の転送先の登録アドレスを示すホップポイントを検索する処理をコンピュータに実行させる検索プログラムが記録された、コンピュータ読み取り可能な記録媒体であって、

前記ホップポイントを検索するにあたり、

前記送信先アドレスを上位から前記一定ビットずつ複数段に分割して得られる各段の分割ビット列値のうち最上段の分割ビット列値をキーとして、前記第一テーブル群のうち最上段の前記第一テーブルを検索し、当該分割ビット列値と一致する前記オフセットアドレスのエントリから前記第二テーブルポインタを読み出す処理と、

前記第二テーブルポインタの示す前記第二テーブルのエントリの前記継続／終了フラグが検索継続を指示している場合には、前記第一テーブルポインタの示す次段の前記第一テーブルを、前記送信先アドレスの当該段の分割ビット列値をキーとして検索し、当該分割ビット列値と

一致する前記オフセットアドレスのエントリから前記第二テーブルポインタを繰返し読み出す処理と、

前記第二テーブルポインタの示す前記第二テーブルのエントリの前記継続／終了フラグが検索終了を指示している場合には、当該エントリの前記ホップポイントを読み出して検索を終了する処理とをコンピュータに実行させる検索プログラムが記録された記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、ネットワーク上のルータに転送されてきた IP (internet protocol) パケット等のビット列のデータ信号の次の転送先を、そのデータ信号の有する IP アドレス等の送信先アドレスに基づいて検索する技術に関する。

【0002】

【従来の技術】通信ネットワークにおける IP パケット信号等のデータ信号の送出先を示すルート情報を検索する検索装置の一つとして、ルータがある。このルータは、データ信号が入力されると、検索テーブルを検索してデータ信号の送出ルートの振り分けを行う。

【0003】この検索テーブルには、IP パケット信号の次の転送先としての「出力インターフェイス」又は「次の中継先であるルータの IP アドレス」が、IP パケット信号の有する「IP サブネットアドレス」及び「プレフィックス長」と対応づけられて登録されている。

【0004】そして、ルータは、受信した IP パケット信号の送信先を示す IP アドレスを検索キーとして検索テーブルの検索を行い、「出力インターフェイス」等の転送先を決定する。この検索を行うにあたっては、検索テーブルに登録されたものの中から、最終的にロングストマッチ法により、この IP アドレスと最も一致する「IP サブネットアドレス」を決定する。

【0005】ここで、ロングストマッチ法について説明する。ロングストマッチ法においては、登録されている IP アドレスのうち、検索キーの IP アドレスと一致する有効ビット列が最も長いものを選択する。以下、例を挙げて説明する。この例では、ビット長が 3 ビットの IP アドレスとして、「0**」及び「00*」の二つが登録されているとする。

【0006】まず、「0**」は、先頭ビットの「0」のみが有効ビット値で、2 番目以降の「**」は、

「0」又は「1」の任意の値を表す。また、「00*」は、1 ビット目及び 2 ビット目の「00」が有効ビット値で、3 番目の「*」は、「0」又は「1」の任意の値を表す。なお、アドレスのうち有効ビット列の長さをビット数で表したものをプレフィックス長（有効ビット列長）という。

【0007】そして、転送されてきた IP パケットの最終的な送信先を示す、検索キーとしての IP アドレス

と、登録されている二つの登録IPアドレスとをそれぞれ照合する。検索キーのIPアドレスを「0001…」とした場合、「00*」の方が「0**」よりも、登録IPアドレスと一致するプレフィックス長が長い。したがって、この場合、次の転送先のIPアドレスとして、「00*」が選択される。

【0008】ところで、IPアドレスのビット数が、上述した例のように3ビット程度と短い場合には、その検索も容易である。しかし、実際に使用されるIPアドレスのビット数は、これよりも遥かに長い。例えば、標準規格IPv4におけるアドレス長は、32ビットである。さらに、近年標準化されたIPv6においては、アドレス長が128ビットと一層長くなっている。このため、ルータにおけるデータ信号中継処理においては、検索処理の効率化が重要となる。

【0009】そこで、従来から、IPパケット信号の次の転送先を、そのIPパケット信号の有するIPアドレスに基づいて検索する方法が、種々提案されている。ここでは、まず、図13を参照して、バイナリ・ツリー(Binary Tree) 検索法について簡単に説明する。

【0010】図13は、バイナリ・ツリー検索法を説明するための概念図である。図13に示す例では、登録されたP1～P8の各IPアドレスがツリー構造を構成している。そして、バイナリ・ツリー検索法は、1ビット単位でビット列のビット値の比較を順次に行う。このため、図13に示すように、ツリーを構成する各ノードは、ビット列の各ビット値にそれぞれ対応する。

【0011】しかし、バイナリ・ツリー検索法では、エントリ数が増えると、ノードの数が急増する。その結果、検索回数も急増してしまう。そこで、ノードの増加に伴う検索回数の増加を抑制する検索方法として、ラディックス・ツリー(Radix Tree) 検索法が提案されている。

【0012】次に、図14を参照して、このラディックス・ツリー検索法について簡単に説明する。図14は、ラディックス・ツリー検索法を説明するための概念図である。図14に示す例では、登録されたP1～P8の各IPアドレスがツリー構造を構成する。そして、ラディックス・ツリー検索法では、バイナリ・ツリー検索法と異なり、ツリー構造のノードを1ビット単位で配置する必要がない。このため、ラディックス・ツリー検索法におけるノード数は、エントリ数(登録されたIPアドレス数)の高々二倍に抑えられる。

【0013】しかし、ラディックス・ツリー検索法では、ツリー構造の枝の「0」及び「1」が直接ビット列のビット値を表しているわけではない。このため、各ノードにおいて、ノードのエントリと検索キーとを比較する必要がある。また、ラディックス・ツリー検索法では、登録されたIPアドレスのビット列の分布によって検索回数が異なる。そして、検索回数は、最大、ビット

列の長さと同じ回数となってしまうことがある。例えば、割当てビット数が32ビットのIPv4においては、検索回数が最大32回となることがある。

【0014】そこで、ノードの増加に伴う検索回数の増加を一層抑制する検索方法の一例が、文献1:「エーシーエム・シグメトリーズ'98、1998年6月、第1～10頁(ACM SIGMETRICS'98, pp.1-10, June 1998)」に、コントロールド・プレフィックス・エクスパンション(Controlled Prefix Expansion) 検索法(以下、「拡張法」と称する。)として提案されている。なお、この拡張法と同様の検索方法は、文献2:「特開平10-257066号公報」にも開示されている。

【0015】次に、図15を参照して、従来の拡張法について説明する。拡張法では、IPアドレスのビット列を複数の分割ビット列に分割し、階層化された分割ビット列ごとの検索テーブル(文献2においては「制御テーブル」。)を設ける。そして、各検索テーブルにおいては、各分割ビット列をそれぞれエントリアドレスとする。

【0016】すなわち、図15に示した例では、合計7ビット分のビット列を、順次に2ビット分、3ビット分及び2ビット分の三つの分割ビット列に分割している。そして、一段目及び三段目の検索テーブルでは、「00」～「11」の四つの分割ビット列をそれぞれエントリアドレスとしている。また、二段目の検索テーブルでは、「000」～「111」の八つの分割ビット列をエントリアドレスとしている。

【0017】そして、拡張法では、登録アドレスを更新する際に、終了/継続フラグ、次の転送先のIPアドレスなどを示す次テーブルポインタや送信インターフェイスの情報そのものを、その検索テーブルの各エントリのうち、登録し得る全てのエントリにそれぞれ設定する。

【0018】すなわち、IPアドレスの登録にあたっては、各検索テーブルにおいて、各分割ビット列のビット長に満たないエントリのビット列を各分割ビット列の末端まで拡張する。ビット列を拡張すると、拡張前のビット列を共有する複数のエントリが、共通の登録IPアドレスに対応することになる。

【0019】図15に示した例では、P1～P8の登録される各IPアドレスをそれぞれ、各分割ビット列の最下位ビットまで拡張する。例えば、プレフィックス長が「1」であるP5のIPアドレス「0*」は、1ビット分だけ拡張されて、2ビットの「00」及び「01」となる。そして、一段目の検索テーブルにおいて、エントリアドレスが「00」及び「01」の両方のエントリに、P5がそれぞれ登録される。

【0020】また、例えば、プレフィックス長が「3」であるP2のIPアドレス「111*」は、2ビット分だけ拡張されて、5ビットの「11100」、「11101」、「11110」及び「11111」の四つのエ

ントリとなる。そして、二段目の検索テーブルにおいては、3ビット目～5ビット目の分割ビット列がエントリアドレスとなる。したがって、エントリアドレスが、「100」、「101」、「110」及び「111」の四つのエントリに、P2がそれぞれ登録される。さらに、他の登録IPアドレスも同様に拡張して登録される。

【0021】このように、ビット列を拡張することにより、登録IPアドレスを検索テーブルの特定のエントリアドレスと対応づけることができる。その結果、各段の検索テーブルの検索にあたり、登録IPアドレスの分割ビット列をそのままオフセットとして使用できる。

【0022】したがって、上述したラディックス・ツリー検索で必要であった各ノードでのエントリとの比較が不要となる。そして、検索回数は、最大でも、検索テーブルの段数で済む。例えば、図15に示した例では、三段の検索テーブルを設けているので、検索回数は最大でも三回で済む。これにより、拡張法によれば、検索回数を大幅に減らすことができる。

【0023】さらに、検索キーのIPアドレスに対応するIPアドレスが未登録の場合には、検索テーブルにその検索キーのIPアドレスを登録することもできる。ここで、図16及び図17を参照して、32ビットのIPアドレスを8ビットずつ四つの分割ビット列に分割した場合の、拡張法によるIPアドレスの登録方法について説明する。

【0024】図16及び図17は、IPアドレスの登録方法を説明するためのフローチャートである。図16では、登録しようとするIPアドレスのプレフィックス長が8ビット以下の場合である場合に、IPアドレスを登録する手順を示している。この場合には、一段目の検索テーブル（第一制御テーブル）に対してのみ、登録することとなる。

【0025】すなわち、登録にあたっては、まず、一段目の検索テーブルのエントリのうち、そのIPアドレスを登録し得る範囲を決定する（図16のS1）。例えば、登録しようとするIPアドレスが、プレフィックス長「1」の「8.0.0.0」である場合について説明する。この場合、エントリアドレスのビット列うち、1ビット目が「1」である上位8ビットの全ての分割ビット列が登録し得る範囲となる。すなわち、「128（＝1000 0000）.0.0.0」～「255（＝1111 1111）」の各エントリが登録し得る範囲となる。

【0026】また、例えば、登録しようとするIPアドレスが、プレフィックス長「2」の「8.0.0.0」である場合について説明する。この場合、エントリアドレスのビット列のうち、1ビット目及び2ビット目が「10」である上位8ビットの全ての分割ビット列が登録し得る範囲となる。すなわち、「128（＝1000

0000）.0.0.0」～「191（＝1011 1111）.0.0.0」の各エントリが登録し得る範囲となる。

【0027】次に、登録し得る範囲に対して、更新制御を行う（図16のS2）。具体的には、図16のS1のステップにおいて決定した登録し得る範囲内のエントリに、既にIPアドレスが登録済みのものが無い場合には、新しいIPアドレスにその範囲内のエントリを更新し、送信インターフェイスも更新して、登録を終了する。

【0028】また、その範囲内に、既に登録済みのIPアドレスがある場合には、まず、登録済みのIPアドレスのプレフィックス長と新たに登録しようとするIPアドレスのプレフィックス長とを比較する。そして、登録済みのIPアドレスのプレフィックス長が新たなプレフィックス長以上である場合には、登録済みのままとして、登録を終了する。一方、これに対して、登録済みのIPアドレスのプレフィックス長が新たなプレフィックス長よりも短い場合には、その範囲のエントリの内容を、新しく登録するIPアドレスの情報に更新し、送信インターフェイスも更新して、登録を終了する。

【0029】また、図17では、登録しようとするIPアドレスのプレフィックス長が32ビット中、8ビットより長く、16ビット以下の場合である場合に、IPアドレスを登録する手順を示している。この場合には、一段目の検索テーブル（第一制御テーブル）及び二段目の検索テーブル（第二制御テーブル）に対してのみ、登録することとなる。

【0030】すなわち、登録にあたっては、まず、図16のS1のステップと同様に、最上位の8ビットの分割ビット列値をキーとして、第一制御テーブルを検索する（図17のS1）。次に、その第一制御テーブル中の該当ビットでの、登録済みのエントリの有無を判断する（図17のS2）。

【0031】そして、登録済みのエントリが有る場合には、その登録済みエントリが示す第二制御テーブルへアクセスする（図17のS3）。一方、登録済みのエントリが無い場合には、新規の第二制御テーブルを、新規の次テーブルポインタにより選択し、設定する（図17のS4）。続いて、その新規の第二制御テーブルへアクセスする（図17のS5）。

【0032】さらに、第二制御テーブルで、図16のS1のステップと同様にして、登録し得る範囲を決定する（図17のS6）。続いて、登録し得る範囲に対して、図16のS2のステップ同様にして、更新制御を行い（図17のS7）、登録を終了する。

【0033】このようにして、従来の拡張法においては、次の転送先に送るための必要な新たな情報が追加された場合、これら送信インターフェイス等の情報に加えて、この追加情報そのものも、登録し得る全てのエント

りに設定していた。また、登録するIPアドレスのプレフィックス長が16ビットより長く24ビット以下の場合や、24ビットよりも長く32ビット以下の場合も、同様にして、エントリの設定を行うことができる。なお、登録済みのIPアドレスを削除する場合も、追加する場合と同様にして、検索テーブルの該当する全てのエントリに登録されていた送信インターフェイスなどの情報を削除していた。

【0034】

【発明が解決しようとする課題】しかしながら、上述した拡張法では、検索テーブルが複数段に分割されている上、一つの登録アドレスが、ビット列を拡張して複数のエントリに登録されているため、登録アドレスの変更（追加、更新及び削除を含む。）が複雑になるという問題点がある。

【0035】すなわち、拡張法では、一つのIPアドレスを登録したり削除したりするのに、通常、複数のエントリの内容を変更しなければならない。例えば、図15に示した例では、登録アドレス「P2」の登録を削除するにあたり、二段目の検索テーブルの「100」、「101」、「110」及び「111」の四つのエントリにおいてそれぞれ登録アドレス「P2」を削除しなければならない。

【0036】また、拡張法では、新たに登録しようとするIPアドレスに対応する段の検索テーブルがない場合には、一つのIPアドレスを登録するために、わざわざ一段分の検索テーブルを追加しなければならない。その上、その検索テーブルの該当する各エントリに、送信インターフェイスなどの同一の登録情報をいちいち書込む必要も生じる。

【0037】さらに、拡張法では、二段目以降の各検索テーブルは、通常、直前の段の検索テーブルの各登録エントリに個別に対応するように複数個ずつ設けられる。その結果、多数の検索テーブルが必要となるため、検索テーブルを構成するためのメモリ容量が大きくなる。その上、拡張法では、送信先アドレスの登録されたエントリが一つの検索テーブルに一つしかない場合であっても、一段分の検索テーブルを構成する必要があった。このため、拡張法では、メモリ容量を浪費してしまうという問題点があった。

【0038】なお、メモリ容量を節約するために、検索テーブルを可変長として、エントリの数に合わせて可変長とすることも考えられる。しかし、検索テーブルを可変長とすると、検索テーブルの管理及び制御が困難となり、特に、ハードウェアによる実現が困難となる。

【0039】本発明は、上記の問題を解決すべくなされたものであり、メモリ容量が節約でき、かつ、登録アドレスの変更が容易な転送先検索方法、転送先検索装置、検索テーブルが記録された記録媒体、及び、検索プログラムが記録された記録媒体の提供を目的とする。

【0040】

【課題を解決するための手段】この目的の達成を図るため、本発明の請求項1に係る転送先検索方法によれば、転送されてきたデータ信号の次の転送先を決定するために、当該データ信号の送信先アドレスを検索キーとして、検索テーブルから、当該データ信号の転送先の登録アドレスを示すホップポイントを検索する転送先検索方法において、検索テーブルを、第一テーブル群と第二テーブル群とにより構成しておき、第一テーブル群は、送信先アドレスに割当てられたビット数分のビット列を上位から一定ビットずつ複数段に分割して得られた各段の分割ビット列にそれぞれ対応する複数の第一テーブルからなり、各第一テーブルは、当該第一テーブルの対応する段の分割ビット列で表される各分割ビット列値を、当該第一テーブルの各エントリのオフセットアドレスとしてそれぞれ有するとともに、オフセットアドレスが、登録アドレスのビット列値のうち当該段に対応する分割ビット列値と一致する場合に、当該第一テーブルは、当該オフセットアドレスのエントリ内容として、次にアクセスする第二テーブルのエントリを示す第二テーブルポイントを有し、かつ、登録アドレスの有効ビット列の最下位ビットが分割ビット列の途中までしかない場合に、当該第一テーブルは、当該有効ビット列のうちの当該分割ビット列に相当する有効ビット列部分を共通して含む各オフセットアドレスのエントリ内容として、互いに共通の第二テーブルポイントを有し、第二テーブルは、エントリ内容として、検索継続又は終了を指示する継続／終了フラグ用フィールド、ホップポイント用フィールド、及び、次にアクセスする第一テーブルを示す第一テーブルポイント用フィールドを有し、ホップポイントを検索するにあたり、送信先アドレスのビット列値を上位から前記一定ビットずつ複数段に分割して得られる各段の分割ビット列値のうち最上段の分割ビット列値をキーとして、第一テーブル群のうち最上段の第一テーブルを検索し、当該分割ビット列値と一致するオフセットアドレスのエントリから第二テーブルポイントを読み出し、第二テーブルポイントの示すエントリの継続／終了フラグが検索継続を指示している場合には、第一テーブルポイントの示す次段の第一テーブルを、送信先アドレスの当該段の分割ビット列値をキーとして検索し、当該分割ビット列値と一致するオフセットアドレスのエントリから第二テーブルポイントを繰返し読み出し、第二テーブルポイントの示すエントリの継続／終了フラグが検索終了を指示している場合には、当該エントリのホップポイントを読み出して検索を終了する方法としてある。

【0041】このように、本発明によれば、階層化された第一テーブルの他に、インデックスとしての第二テーブルを設ける。そして、第二テーブル、第一テーブルと第二テーブルとを交互に検索する。そのために、第二テーブルのエントリには、第一テーブルポイントや、ホッ

ブポイントなどの転送先に関するデータを格納しておく。その結果、膨大な数の各第一テーブルの各エントリには、原則、第二テーブルポイントのみを格納しておくだけでよい。すなわち、各第一テーブルに、ホップポイントなどの転送先に関する多量のデータをいちいち格納する必要がない。

【0042】これにより、本発明においては、個々の第一テーブルを構成する個々のメモリ容量を、従来の拡張法における個々の検索テーブルのメモリ容量よりも少なくすることができる。その結果、膨大な数の第一テーブルからなる第一テーブル群全体のメモリ量を大幅に節約することができる。したがって、本発明によれば、従来の拡張法よりも検索テーブル全体のメモリ容量を節約することができる。

【0043】また、請求項2記載の発明によれば、第二テーブルの各エントリは、ホップポイント更新の有無を指示する更新フラグ用フィールドを有し、ホップポイントを検索するにあたり、初期値のホップポイントを設定して、内部変数として保持し、第二テーブルポイントの示す第二テーブルのエントリの更新フラグが更新を指示している場合に、内部変数として保持されてホップポイント、当該エントリから読み出したホップポイントに更新する方法としてある。このように、第二テーブルに更新フラグを設ければ、検索中に必要に応じて、ホップポイントを更新することができる。

【0044】また、請求項3記載の発明によれば、第一テーブルのエントリは、ホップポイントを更新する必要がある場合にのみ、第二テーブルポイントの代わりに、登録アドレスのビット列値に従って次にアクセスする次段の第一テーブルを示す第一テーブルポイントの示す第一テーブルを検索するにあたり、第一テーブルのエントリに第一テーブルポイントが設定されている場合には、第二テーブルを経由せずに、テーブルポイントの示す第一テーブルへ直接アクセスする方法としてある。

【0045】このように、第二テーブルポイントの代わりに次段の第一テーブルを示すテーブルポイントの示す第一テーブルへアクセスせずに、次段の第一テーブルへアクセスすることができる。その結果、検索中のアクセス回数を減らすことができるので、検索の高速化を図ることができるとともに、そのテーブルポイントの分だけ、第二テーブルのエントリ数を減らすことができるので、第二テーブルのメモリ容量の節約を図ることもできる。

【0046】また、請求項4記載の発明によれば、第一テーブルを、0ビット用第一テーブルと1ビット用第一テーブルと分けて構成しておき、0ビット用第一テーブルは、第一テーブルの対応する分割ビット列の先頭ビットの値が「0」である場合の、当該分割ビット列から当該先頭ビットを除いた残存分割ビット列に対応し、1ビ

ット用第一テーブルは、第一テーブルの対応する分割ビット列の先頭ビットの値が「1」である場合の、当該分割ビット列から当該先頭ビットを除いた残存部分ビット列に対応し、第二テーブルは、各エントリに、0ビット用第一テーブルを示す0ビット用第一テーブルポイント用フィールドと、1ビット用第一テーブルを示す1ビット用第一テーブルポイント用フィールドとを有し、ホップポイントを検索するにあたり、分割ビット列の先頭ビット値が「0」であるときに、0ビット用第一テーブルポイントの示す0ビット用第一テーブルにアクセスし、分割ビット列の先頭ビット値が「1」であるときに、1ビット用第一テーブルポイントの示す1ビット用第一テーブルにアクセスする方法としてある。

【0047】このように、各段の第一テーブルをそれぞれ0ビット用及び1ビット用第一テーブルとに分けておけば、該当する登録エントリがない場合には、0ビット用又は1ビット用のいずれか一方の第一テーブルを省略することができる。その結果、第一テーブルのメモリ容量を節約することができる。さらに、分割ビット列の先頭ビットに応じて、0ビット用又は1ビット用のいずれか一方の第一テーブルのみを選択してアクセスすることができる。その結果、第一テーブルの検索エントリ数を半分とすることができる。

【0048】また、請求項5記載の発明によれば、第一テーブルが、登録アドレスに対応するエントリ（登録エントリ）を一つだけ有する場合に、当該第一テーブルを当該エントリ一つだけのメモリ領域の記憶容量を有する単独第一テーブルとし、第二テーブルの各エントリは、当該エントリの第一テーブルポイントの示す第一テーブルの種類が単独第一テーブルであるか否かを示す第一テーブル種類フラグ用フィールドを有し、ホップポイントを検索するにあたり、第二テーブルポイントの示す第二テーブルのエントリのテーブル種類フラグが、当該エントリの第一テーブルポイントが示す第一テーブルが当該単独第一テーブルであることを示している場合に、当該単独第一テーブルに対応する登録アドレスの分割ビット列値と、送信先アドレスの有効ビット列のうち当該分割ビット列に相当する有効ビット列部分とを照合し、当該分割ビット列値と当該有効ビット列部分とが不一致の場合に、検索を終了し、当該分割ビット列値と当該有効ビット列部分とが一致した場合に、当該エントリの第二テーブルポイントの示す第二テーブルのエントリへアクセスする方法としてある。

【0049】一般に、送信先アドレスに割り当てられたビット列の長さに対して、登録アドレスの数が少ない場合には、一つの第一テーブルに登録されるエントリ数が少なくなる。この場合、一つの第一テーブルあたりの登録エントリ数が一つだけとなる場合が少なからず発生する。このような場合、たった一つの登録エントリのために、分割ビット列分の全エントリを有する第一テーブル

のメモリ領域を確保することは、メモリ容量の浪費となる。そこで、一エントリ分だけの単独第一テーブルを設ければ、一つの第一テーブルあたりの登録エントリが一つだけの場合に、メモリ容量を節約することができる。

【0050】また、請求項6記載の発明によれば、単独第一テーブルを省略し、第二テーブルにおいて、当該単独第一テーブルを示す第一テーブルポインタの代わりに、その単独第一テーブルのエントリに設定されるべき第二テーブルポインタを設定する方法としてある。

【0051】このようにすれば、単独第一テーブルを省略することができるので、メモリ容量の一層の節約を図ることができる。また、省略された第一テーブルへのアクセス回数分だけアクセス回数を減らすことができるので、検索の迅速化を図ることができる。

【0052】また、請求項7記載の発明によれば、第二テーブルの先頭エントリに、最上段の第一テーブルの先頭位置アドレスを示す先頭ポインタを格納しておき、ホップポインタの検索を開始するにあたり、第二テーブルの先頭エントリにアクセスし、当該先頭エントリの第一ポインタの示す第一テーブルにアクセスする方法としてある。

【0053】また、請求項8記載の発明によれば、検索テーブルにおいて登録アドレスを変更するにあたり、第二テーブルのエントリにおいて、第一テーブルポインタ及びホップポインタを変更し、第一テーブルのエントリにおいて、第二テーブルポインタを変更する方法としてある。

【0054】このようにすれば、膨大な第一テーブルの該当エントリにおいては、第二テーブルポインタのみを変更すれば良く、ホップポインタなどの転送先に関する情報の変更は、第二テーブルでのみ行えば済む。したがって、本発明によれば、登録アドレスの変更（追加、更新及び削除を含む。）を容易に行うことができる。

【0055】また、本発明の請求項9記載の転送先検索装置によれば、ルータに転送されてきたデータ信号の次の転送先を決定するために、当該データ信号の送信先アドレスを検索キーとして、検索テーブルから、当該データ信号の転送先の登録アドレスを示すホップポインタを検索する転送先検索装置において、検索テーブルを構成する第一テーブル群を格納した第一記憶部と、検索テーブルを構成する第二テーブルを格納した第二記憶部と、第一テーブル群へのアクセスを制御する第一制御部と、第二テーブルへのアクセスを制御する第二制御部とを備え、第一テーブル群は、送信先アドレスに割当てられたビット数分のビット列を上位から一定ビットずつ複数段に分割して得られた各段の分割ビット列にそれぞれ対応する複数の第一テーブルからなり、各第一テーブルは、当該第一テーブルの対応する段の分割ビット列で表される各分割ビット列値を、当該第一テーブルの各エントリのオフセットアドレスとしてそれぞれ有するとともに、

オフセットアドレスが、登録アドレスのビット列値のうち当該段に対応する分割ビット列値と一致する場合に、当該第一テーブルは、当該オフセットアドレスのエントリ内容として、次にアクセスする第二テーブルのエントリを示す第二テーブルポインタを有し、かつ、登録アドレスの有効ビット列の最下位ビットが分割ビット列の途中までしかない場合に、当該第一テーブルは、当該有効ビット列のうちの当該分割ビット列に相当する有効ビット列部分を共通して含む各オフセットアドレスのエントリ内容として、互いに共通の前記第二テーブルポインタを有し、第二テーブルは、エントリ内容として、検索継続又は終了を指示する継続／終了フラグ用フィールド、前記ホップポインタ用フィールド、及び、次にアクセスする前記第一テーブルを示す第一テーブルポインタ用フィールドを有し、第一制御部は、送信先アドレスを上位から一定ビットずつ複数段に分割して得られる各段の分割ビット列値のうち最上段の分割ビット列値をキーとして、第一テーブル群のうち最上段の第一テーブルを検索し、当該分割ビット列値と一致するオフセットアドレスのエントリから第二テーブルポインタを読み出し、かつ、当該第二テーブルポインタを第二制御部へ転送し、第二制御部は、第二テーブルポインタの示す第二テーブルのエントリの継続／終了フラグが検索継続を指示している場合に、第一テーブルポインタを読み出し、かつ、当該第一テーブルポインタを第一制御部へ転送し、第一制御部は、第一テーブルポインタの示す次段の第一テーブルを、送信先アドレスの当該段の分割ビット列値をキーとして検索し、当該分割ビット列値と一致するオフセットアドレスのエントリから第二テーブルポインタを繰返し読み出し、かつ、当該第二テーブルポインタを第二制御部へ転送し、第二制御部は、第二テーブルポインタの示す第二テーブルのエントリの継続／終了フラグが検索終了を指示している場合には、当該エントリのホップポインタを読み出し、当該ホップポインタを次の転送先として出力する構成としてある。

【0056】このように、本発明の転送先検索装置によれば、階層化された第一テーブルの他に、インデックスとしての第二テーブルを設け、第一テーブルと第二テーブルとを交互に検索する。そして、次に転送先を示すホップポインタを第二テーブルにのみ格納し、第一テーブルは、第二テーブルポインタのみをエントリ内容とする。

【0057】これにより、個々の第一テーブルを構成するメモリ容量を、従来の拡張法の個々の検索テーブルのメモリ容量よりも少なくして、検索テーブル全体のメモリ容量を節約することができる。

【0058】また、本発明では、インデックスとしての第二テーブルにのみホップポインタを設けているので、膨大な第一テーブルの該当エントリにおいては、第二テーブルポインタのみを変更すれば良く、ホップポインタ

などの転送先に関する情報の変更は、第二テーブルでのみ行えば済む。したがって、本発明によれば、登録アドレスの変更（追加及び削除を含む。）を容易に行うことができる。

【0059】また、本発明の請求項10記載の検索テーブルの記録された記録媒体によれば、転送されてきたデータ信号の次の転送先を決定するために、当該データ信号の送信先アドレスを検索キーとして、当該データ信号の転送先の登録アドレスを示すホップポインタを検索するための検索テーブルの記録された、コンピュータ読み取り可能な記録媒体であって、検索テーブルは、第一テーブル群と第二テーブルとにより構成され、第一テーブル群は、送信先アドレスに割当てられたビット数分のビット列を上位から一定ビットずつ複数段に分割して得られた各段の分割ビット列にそれぞれ対応する複数の第一テーブルからなり、各第一テーブルは、当該第一テーブルの対応する段の分割ビット列で表される各分割ビット列値を、当該第一テーブルの各エントリのオフセットアドレスとしてそれぞれ有するとともに、オフセットアドレスが、登録アドレスのビット列値のうち当該段に対応する分割ビット列値と一致する場合に、当該第一テーブルは、当該オフセットアドレスのエントリ内容として、次にアクセスする第二テーブルのエントリを示す第二テーブルポインタを有し、かつ、登録アドレスの有効ビット列の最下位ビットが分割ビット列の途中までしかない場合に、当該第一テーブルは、当該有効ビット列のうちの当該分割ビット列に相当する有効ビット列部分を共通して含むに各オフセットアドレスのエントリ内容として、互いに共通の第二テーブルポインタを有し、第二テーブルは、エントリ内容として、検索継続又は終了を指示する継続／終了フラグ用フィールド、ホップポインタ用フィールド、及び、次にアクセスする第一テーブルを示す第一テーブルポインタ用フィールドを有する検索テーブルが、コンピュータ読み取り可能に記録されている。

【0060】このように、本発明によれば、階層化された第一テーブルの他に、インデックスとしての第二テーブルを設け、第一テーブルと第二テーブルとを交互に検索する。そして、次に転送先を示すホップポインタを第二テーブルにのみ格納し、第一テーブルは、第二テーブルポインタのみをエントリ内容とする。これにより、個々の第一テーブルを構成するメモリ容量を、従来の拡張法の個々の検索テーブルのメモリ容量よりも少なくして、検索テーブル全体のメモリ容量を節約することができる。

【0061】また、この発明の請求項11に記載された検索プログラムの記録された記録媒体によれば、転送されてきたデータ信号の次の転送先を決定するために、当該データ信号の送信先アドレスを検索キーとして、検索テーブルから、当該データ信号の転送先の登録アドレス

を示すホップポインタを検索する処理をコンピュータに実行させる検索プログラムが記録された、コンピュータ読み取り可能な記録媒体であって、ホップポインタを検索するにあたり、送信先アドレスを上位から一定ビットずつ複数段に分割して得られる各段の分割ビット列値のうち最上段の分割ビット列値をキーとして、第一テーブル群のうち最上段の第一テーブルを検索し、当該分割ビット列値と一致するオフセットアドレスのエントリから第二テーブルポインタを読み出す処理と、第二テーブルポインタの示す第二テーブルのエントリの継続／終了フラグが検索継続を指示している場合には、第一テーブルポインタの示す次段の前記第一テーブルを、送信先アドレスの当該段の分割ビット列値をキーとして検索し、当該分割ビット列値と一致する前記オフセットアドレスのエントリから第二テーブルポインタを繰返し読み出す処理と、第二テーブルポインタの示す第二テーブルのエントリの継続／終了フラグが検索終了を指示している場合には、当該エントリのホップポインタを読み出して検索を終了する処理とをコンピュータに実行させる検索プログラムが記録された記録媒体。

【0062】本発明の記録媒体に記録されたプログラムをコンピュータに読み込ませて実行させることにより、検索テーブルのためのメモリ容量を節約しつつ、登録アドレスの変更が容易な送信先アドレス検索を実現することが可能となる。

【0063】

【発明の実施の形態】以下、本発明の実施の形態について、図面を参照して説明する。なお、以下の実施形態における送信先アドレス検索のための処理は、プログラムに制御されたコンピュータにより実行される。このプログラムは、例えば、記録媒体により提供される。また、以下の実施形態における検索テーブルは、コンピュータ読み取り可能な記録媒体として提供される。これら記録媒体としては、例えば、磁気ディスク、半導体メモリ、その他の任意の、コンピュータで読み取り可能なものを使用することができる。

【0064】〔第一実施形態〕

（構成について）まず、図1を参照して、第一実施形態の転送先検索装置の構成について説明する。図1は、第一実施形態の転送先検索装置の構成を説明するための機能ブロック図である。

【0065】第一実施形態の転送先検索装置は、ルータに他のルータ等から転送されてきたデータ信号の次の転送先を決定する装置である。そのために、この転送先検索装置は、データ信号に含まれている、当該データ信号の送信先を示す送信先アドレスを検索キーとして、検索テーブルから、そのデータ信号の転送先を示すホップポインタを検索する。

【0066】そのために、第一実施形態の転送先検索装置は、検索テーブルを構成する第一テーブル群10が格

納された第一記憶部30と、検索テーブルを構成する第二テーブル20が格納された第二記憶部40と、第一テーブル群へのアクセスを制御する第一制御部50と、第二テーブルへのアクセスを制御する第二制御部60とにより構成されている。

【0067】なお、第一及び第二記憶部30及び40は、例えば、任意にデータの読み書き可能な記憶ディスクや半導体メモリその他の、任意好適な記録装置により構成すると良い。また、第一及び第二記憶部30及び40は、一つの記憶装置の記憶領域に個別に設けても良

い。

【0068】(検索テーブルについて)ここで、図2を参照して、検索テーブルを構成する第一テーブル群10及び第二テーブル20について説明する。

【0069】(第一テーブルについて)図2は、第一実施形態の転送先検索方法を説明するための検索テーブルの概念構成図である。図2に示すように、第一テーブル群10は、送信先アドレスに割当てられた32ビット数分のビット列を上位から8ビットずつ4段に分割して得られた各段の分割ビット列にそれぞれ対応する一段目～

四段目第一テーブル11～14からなる。

【0070】すなわち、一段目第一テーブル11は、31ビット目から0ビット目までの32ビットのビット列のうち、最上位の31ビット目から24ビット目まで(以下、「Bits [31-24]」と表記する。)の分割ビット列に対応する。続いて、二段目第一テーブル12は、Bits [23-16]の分割ビット列に対応する。以下同様に、三段目第一テーブル13は、Bits [15-8]の分割ビット列に対応し、四段目第一テーブル14は、Bits [7-0]の分割ビット列に対応する。

【0071】さらに、第一実施形態では、各段の第一テーブルを、それぞれ0ビット用第一テーブルと1ビット用第一テーブルとに分けて構成している。なお、図2では、0ビット用及び1ビット用第一テーブルのいずれか一方のみを、各段の第一テーブル11～14として便宜的に示す。

【0072】そして、0ビット用第一テーブルは、各段の第一テーブル11～14のそれぞれ対応する分割ビット列の先頭ビットの値が「0」である場合の、当該分割ビット列から当該先頭ビットを除いた残存分割ビット列に対応する。例えば、一段目の0ビット用第一テーブルは、Bits [31-24]の分割ビット列のうち、31ビット目の先頭ビット値が「0」である場合のBits [30-24]に対応する。また、二段目以降の0ビット用第一テーブルについても同様である。

【0073】また、1ビット用第一テーブルは、各段の第一テーブル11～14のそれぞれ対応する分割ビット列の先頭ビットの値が「1」である場合の、当該分割ビット列から当該先頭ビットを除いた残存部分ビット列に

対応する。例えば、二段目の1ビット用第一テーブルは、Bits [23-16]の分割ビット列のうち、23ビット目の先頭ビット値が「1」である場合のBits [22-16]に対応する。また、他段の1ビット用第一テーブルについても同様である。

【0074】このように、各段の第一テーブルをそれぞれ0ビット用及び1ビット用第一テーブルとに分けておけば、該当する登録エントリがない場合には、0ビット用又は1ビット用のいずれか一方の第一テーブルを省略することができる。その結果、第一テーブルのメモリ容量を節約することができる。

【0075】そして、各第一テーブル11～14は、当該第一テーブルの対応する段の分割ビット列で表される各分割ビット列値を、当該第一テーブルの各エントリのオフセットアドレスとしてそれぞれ有する。さらに、各段の0ビット用又は1ビット用第一テーブル11～14は、それぞれ、分割ビット列の先頭ビットを除いた残存部分ビット列で表される分割ビット列値を、各エントリをオフセットアドレスとしてそれぞれ有する。したがって、各段の0ビット用又は1ビット用第一テーブルのエントリ数は、元の第一テーブルのエントリ数の半分となる。

【0076】具体的には、一段目の0ビット用又は1ビット用第一テーブル11は、Bits [30-24]の7ビットの分割ビット列で表される分割ビット列値(000000)～(111 1111)をそれぞれオフセットアドレスとして有する0番目～127番目のエントリを有する。また、二段目の0ビット用又は1ビット用第一テーブル12は、Bits [22-16]の7ビットの分割ビット列で表される分割ビット列値(000000)～(111 1111)をそれぞれオフセットアドレスとして有する0番目～127番目のエントリを有する。

【0077】さらに、三段目及び四段目の0ビット用又は1ビット用第一テーブル13及び14も、同様に、分割ビット列値(000 000)～(111 1111)をそれぞれオフセットアドレスとする0番目～127番目のエントリを有する。なお、図2においては、格段の第一テーブル11～14をそれぞれ一つずつ代表して示したが、実際には、二段目以降の各段の第一テーブル12～14は、前段の各登録エントリごとに個別に設けられている。

【0078】すなわち、本実施形態では、二段目の0ビット用又は1ビット用第一テーブル12の最大数は、一段目の0ビット用又は1ビット用第一テーブル11の全エントリ数分の(2の7乗=128個)となる。さらに、三段目の第一テーブル13は、各二段目の第一テーブル12の登録エントリごとに個別に設けられる。したがって、三段目の第一テーブル13の最大数は、128の2乗個となる。同様にして、四段目第一テーブル14の最

大数は、128の3乗個となる。ただし、各第一テーブル11~14では、常に全エントリに送信先アドレスが登録されているわけではない。

【0079】そして、第一テーブルのオフセットアドレスが、登録アドレスのビット列値のうち当該段に対応する分割ビット列値と一致する場合に、当該第一テーブルは、当該オフセットアドレスのエントリ内容として、次にアクセスする第二テーブルのエントリを示す第二テーブルポインタを有する。

【0080】ここで、図3の(A)を参照して、第一テーブルのエントリ内容について説明する。図3は、第一テーブルのエントリ内容を説明するためのデータ構成図である。図3に示す第一テーブルは、例えば、Bits [31-24]のBit 31のビット値が「0」である場合の、一段目の0ビット用第一テーブル11に相当する。このため、この第一テーブル11は、最上段の8ビットの分割ビット列のうち、Bit 32を除いた残りのBits [30-24]に対応する。このため、この0ビット用第一テーブル11のエントリ数は128個(2の(8-1)乗個)となる。そして、0番目~127番目の各エントリのアドレスは、それぞれ(000 0000)~(111 1111)のオフセットアドレスに対応する。

【0081】また、登録アドレスの有効ビット列の最下位ビットが分割ビット列の途中までしかない場合に、当該第一テーブルは、従来の拡張法と同様に、当該有効ビット列のうちの当該分割ビット列に相当する有効ビット列部分を共通して含む各オフセットアドレスのエントリ内容として、互いに共通の第二テーブルポインタを有する。

【0082】ここで、図3を参照して、登録アドレスのビット列を拡張して登録する場合について説明する。図3は、ビット列を拡張して登録された第一テーブルのエントリを説明するためのデータ構成図である。図3では、二つの登録アドレスのビット列を分割ビット列に合わせて拡張してそれぞれ登録する例について説明する。

【0083】まず、一つ目の登録アドレス「R1」の一段目に対応する分割ビット列Bits [31-24]が、(00** ***)である場合について説明する。この登録アドレス「R1」の有効ビット列の最下位ビットは、30番目のビットであり、分割ビット列の途中である。そして、29番目以下の6ビット分のビット値が不定となっている。したがって、この登録アドレスを拡張すると、(0000 0000)から(0011 1111)までの64通り(2の6乗通り)の分割ビット列値が、有効ビット列部分(00)を共通して含むことになる。

【0084】そこで、図3に示すように、一段目の0ビット用第一テーブルの全エントリのうち、(000 0000)~(011 1111)をオフセットアドレス

とする0番目~63番目の各エントリに、R1用の第二テーブルポインタを設定する。

【0085】次に、二つ目の登録アドレス「R2」の一段目に対応する分割ビット列Bits [31-24]が、(01** ***)である場合について説明する。この登録アドレス「R2」の有効ビット列の最下位ビットは、30番目のビットであり、分割ビット列の途中である。そして、29番目以下の6ビット分のビット値が不定(*)となっている。したがって、この登録アドレスを拡張すると、(0100 0000)から(0111 1111)までの64通り(2の6乗通り)の分割ビット列値が、有効ビット列部分(01)を共通して含むことになる。

【0086】そこで、図3に示すように、一段目の0ビット用第一テーブルの全エントリのうち、(100 0000)~(111 1111)をオフセットアドレスとする64番目~127番目の各エントリに、R2用の第二テーブルポインタを設定する。なお、登録エントリ以外のエントリは、空欄(NUL)となる。

【0087】(第二テーブルについて)次に、図4を参照して、第二テーブルのエントリの内容について説明する。図4では、第二テーブル20の一つのエントリ2の内容を代表して示す。図4に示すように、この第二テーブル20の各エントリは、次に示す九つのフィールドを有する。すなわち、継続/終了フラグ用フィールド21、更新フラグ用フィールド22、ホップポインタ用フィールド23、0ビット用テーブル種類フラグ用フィールド24、0ビット用第一テーブルポインタ用フィールド25、0ビット用単独第一テーブルの登録アドレス及びそのプロフィックス長用フィールド26、1ビット用テーブル種類フラグフィールド27、1ビット用第一テーブルポインタ用フィールド28、及び、1ビット用単独第一テーブルの登録アドレス及びそのプロフィックス長用フィールド29を有する。

【0088】この継続/終了フラグ用フィールド21には、検索の継続又は終了を指示する継続/終了フラグが設定される。また、更新フラグ用フィールド22には、ホップポインタの更新の有無を指示する更新フラグが設定される。さらに、ホップポインタ用フィールド23には、データ信号の次の転送先のアドレスの格納位置を示すポインタが設定される。

【0089】さらに、この第二テーブル20では、各エントリのフィールドを、分割ビット列の先頭ビット値が「0」である場合と、「1」である場合とで分けている。すなわち、0ビット用及び1ビット用テーブル種類フラグ用フィールド24及び27には、当該エントリの第一テーブルポインタの示す第一テーブルの種類が、後述の第二実施形態で説明する単独第一テーブルであるか否かを示すテーブル種類フラグが設定される。

【0090】また、0ビット用及び1ビット用第一テー

ブルポインタ用フィールド 25 及び 28 は、それぞれ登録アドレスのビット列値に従って次にアクセスする 0 ビット用及び 1 ビット用第一テーブルを示す第一テーブルポインタが設定される。さらに、0 ビット用及び 1 ビット用単独第一テーブルの登録アドレス及びそのプロフィックス長用フィールド 26 及び 29 には、第一テーブルポインタが示す第一テーブルが、単独第一テーブルである場合に、単独第一テーブルの単独エントリに登録されている登録アドレスの分割ビット列値とそのプロフィックス長とが設定される。

【0091】その上、第一実施形態では、第二テーブル 20 の先頭エントリに、最上段の第一テーブル 11 の先頭位置アドレスを示す先頭ポインタ格納する先頭ポインタ用フィールド 2a を設けている。

【0092】（動作例）次に、図 2 を参照して、第一実施形態の転送先検索装置の動作例、すなわち、転送先検索方法の一例について説明する。まず、ルータ（図示せず）へ、IP v4 の IP パケット信号、すなわち、32 ビットの送信先アドレスを有するデータ信号が転送されて来たとする。この送信先アドレスは、まず、ルータ内の第二制御部 60 へ入力される。

【0093】続いて、送信先アドレスの転送先を示すホップポインタの検索にあたり、第二制御部 60 が、第二記憶部 40 に格納されている第二テーブル 20 の先頭エントリにアクセスする（図 2 の矢印 a）。

【0094】そして、送信先アドレスの先頭ビット（Bit (31)）のビット値が「0」である場合には、第二制御部 60 は、第二テーブル 20 から当該先頭エントリの示す 0 ビット用第一テーブルポインタを読み出す。この先頭エントリの 0 ビット用第一テーブルポインタは、第一記憶部 30 における一段目の 0 ビット用第一テーブルポインタの先頭エントリ位置（ベースアドレス）を示す。

【0095】また、送信先アドレスの先頭ビット値が「1」である場合には、第二制御部 60 は、1 ビット用第一テーブルポインタを読み出す。この先頭エントリの 1 ビット用第一テーブルポインタは、第二記憶部 40 における一段目の 1 ビット用第一テーブルポインタの先頭エントリ位置（ベースアドレス）を示す。なお、以下、説明を簡単にするために、0 ビット用と 1 ビット用とを特に区別する必要がある場合を除いて、0 ビット用及び 1 ビット用第一テーブルのいずれも単に第一テーブルと称する。

【0096】さらに、第二制御部 60 は、ホップポインタの初期値として「NULL」を読み出す。そして、第二制御部 60 は、読み出した第一テーブルポインタと、内部変数としてのホップポインタとを送信先アドレスとともに、第一制御部 50 へ転送する。なお、第二テーブル 20 の先頭エントリは、第二テーブル 20 へのメモリアクセスを省略するために第二制御部 60 に予め設けお

ても良い。

【0097】次に、第一制御部 50 は、読み出された第一テーブルポインタの示す一段目第一テーブル 11 の先頭エントリにアクセスする（図 2 の矢印 b）。第一制御部 50 は、32 ビットの送信先アドレスを上位から 8 ビットずつ 4 段に分割する。続いて、分割の結果得られる各段の分割ビット列値のうち、最上段の分割ビット列値を検索キーとする。そして、この検索キーにより、一段目第一テーブル 11 のオフセットアドレスを検索する（図 2 に矢印 c）。

【0098】検索の結果、第一制御部 50 は、検索キーの分割ビット列値と一致するオフセットアドレスを有するエントリから第二テーブルポインタを読み出す。さらに、第一制御部 50 は、読み出した第二テーブルポインタを第二制御部 60 へ転送する。

【0099】ここで、図 5 を参照して、第二テーブルポインタを受け取った第二制御部 60 の動作例について説明する。図 5 は、第二制御部 60 の第二テーブル 20 にアクセス後の動作例を説明するためのフローチャートである。図 5 に示すように、第二制御部 60 は、まず、第二テーブルポインタの示す第二テーブル 20 のエントリへアクセスする（図 2 の矢印 d）（図 5 の S1）。

【0100】次に、第二制御部 60 は、第二テーブル 20 のエントリの更新フラグが、更新を指示しているか否かを判断する（図 5 の S2）。そして、更新フラグが更新を指示している場合に、第二制御部 60 は、内部変数として保持しているホップポインタを、そのエントリの有する新しいホップポインタに更新する（図 5 の S3）。

【0101】また、更新フラグが更新を指示していない場合、または、ホップポインタ更新後、第二制御部 60 は、そのエントリの継続／終了フラグが、継続を指示しているか否かを判断する（図 5 の S4）。そして、継続／終了フラグが、終了を指示している場合、第二制御部 60 は、その段階で内部変数として保持しているホップポインタを、次の転送先を示すホップポインタとして出力する。

【0102】また、そのエントリの継続／終了フラグが検索継続を指示している場合、第二制御部 60 は、続いて、分割ビット列の先頭ビット値が「0」であるか「1」であるかを判断する。ここでは、二段目の分割ビット列の先頭ビットである Bit (24) の値を判断する。（図 5 の S5）。なお、三段目の場合には Bit (15)、四段目の場合には Bit (7) の値をそれぞれ判断する。そして、Bit (24) の値が「0」である場合、第二制御部 60 は、そのエントリから、0 ビット用第一テーブルポインタを読み出す（図 5 の S6）。また Bit (24) の値が「1」である場合、第二制御部 60 は、そのエントリから、1 ビット用第一テーブルポインタを読み出す（図 5 の S9）。

【0103】次に、第二制御部60は、そのエントリのテーブル種類フラグにより、読み出した第一テーブルポインタが示す第一テーブルが、単独第一テーブルか否かを判断する(図5のS7又はS10)。そして、単独第一テーブルでない場合には、第二制御部60は、その第一テーブルポインタを第一制御部50へ転送する(図5のS12)。

【0104】また、テーブル種類フラグが単独第一テーブルである場合には、後述する第二実施形態に示すように、第二制御部60は、そのエントリのフィールドからプレフィックス長を読み出す。そして、その分の分割ビット列値の上位からプレフィックス長分のビット列値部分が、単独第一テーブルに対応する登録アドレスの分割ビット列値と一致するか否かを判断する(図5のS8又はS11)。そして、一致した場合、第二制御部60は、その第一テーブルポインタを第一制御部50へ転送する(図5のS12)。なお、一致しなかった場合、第二制御部60は、検索失敗として、検索を終了する。

【0105】第二制御部60から再び第一テーブルポインタが転送された場合、第一制御部50は、第一テーブルポインタの示す二段目の第一テーブルの先頭エントリへアクセスする(図2の矢印e)。そして、一段目の場合と同様にして、送信先アドレスの二段目に対応する分割ビット列値をキーとして二段目第一テーブル12を検索する(図2の矢印f)。そして、その分割ビット列値と一致するオフセットアドレスのエントリから第二テーブルポインタを繰返し読み出し、かつ、当該第二テーブルポインタを第二制御部60へ転送する。

【0106】以下、検索が終了するまで、第二制御部60により第二テーブル20へアクセスし(図2の矢印g)、さらに、第二テーブル20のエントリの第一テーブルポインタの示す三段目第一テーブル13へアクセスし(図2の矢印h)、送信先アドレスの三段目に対応する分割ビット列値と一致するオフセットアドレスを検索し(図2の矢印i)、続いて、そのオフセットアドレスのエントリの第二テーブルポインタの示す第二テーブル20のエントリへアクセスする(図2の矢印j)。

【0107】そして、三段目第一テーブル13を検索しても検索が終了しない場合には、さらに、第二テーブル20のエントリにおいて第一テーブルポインタの示す四段目第一テーブル14へアクセスする(図2の矢印k)。そして、送信先アドレスの四段目に対応する分割ビット列値と一致するオフセットアドレスを検索し(図2の矢印l)、続いて、そのオフセットアドレスのエントリの第二テーブルポインタの示す第二テーブル20のエントリへアクセスする(図2の矢印m)。さらに、最終的に、第二テーブルポインタの示す第二テーブルのエントリの継続/終了フラグが検索終了を指示している場合に、第二制御部60は、当該エントリのホップポインタを読み出し、当該ホップポインタを次の転送先として

出力する(図2の矢印n)。以上のようにして、次の転送先を示すホップポインタを検索することができる。

【0108】次に、図6を参照して、更新フラグ用フィールド21の更新フラグの指示による、ホップポインタの更新について詳細に説明する。ここでは、従来のバイナリ・ツリー検索法の説明の際に用いた図13に示した登録されたP1~P8の各IPアドレスにより構成されたツリー構造と同一の構成例を用いる。

【0109】そして、図6では、IPアドレスを2ビットずつに分割した例について説明する。すなわち、一段目第一テーブルは、1及び2ビット目に対応し、二段目第一テーブルは3及び4ビット目に対応し、三段目第一テーブルは5及び6ビット目に対応し、さらに、四段目第一テーブルは、7及び8ビット目に対応するものとする。

【0110】そして、ここでは、P8のIPアドレスを検索する場合の、途中のホップポインタの変遷について説明する。まず、第二テーブルの先頭エントリに、初期のホップポインタを設定しておく。また、一段目第一テーブルで検索した第二テーブルポインタの示す第二テーブルのエントリに、更新を指示する更新フラグとともに、新たなホップポインタとして、P6のIPアドレスを示すホップポインタを設定しておく。また、二段目第一テーブルで検索した第二テーブルポインタの示す第二テーブルのエントリに、更新を指示する更新フラグとともに、新たなホップポインタとして、P7のIPアドレスを示すホップポインタを設定しておく。

【0111】また、三段目第一テーブルで検索した第二テーブルポインタの示す第二テーブルのエントリに、更新を指示する更新フラグとともに、新たなホップポインタとして、P8のIPアドレスを示すホップポインタを設定しておく。このように、第二テーブルのエントリにホップポインタを設定しておくことにより、ホップポインタが、初期設定から、P1、P6及びP7を示すホップポインタに順次に更新されながら、最終的にP8のホップポインタが検索される。

【0112】一方、例えば、図6に示すP6のIPアドレスが登録されていない場合は、その段でホップポインタを更新する必要が無い。このため、二段目第一テーブルで検索した第二テーブルポインタの示す第二テーブルのエントリに、更新不指示の更新フラグを設定しておく、ホップポインタを空欄(NULL)としておく。この場合、ホップポインタは、P6を飛ばして、初期設定から、P1、P7及びP8と順次に更新されることになる。

【0113】(検索テーブルの変更)本発明においては、第一テーブルのエントリには、原則、第二テーブルポインタだけが設定されている。このため、検索テーブルの登録アドレスを変更するにあたっては、第二テーブルのエントリで第一テーブルポインタ及びホップポイン

タを変更するとともに、第一テーブルのエントリにおいては、第二テーブルポインタのみを変更するだけで良い。したがって、ホップポインタなどの転送先に関する実際の情報の変更は、第二テーブルでのみ行えば済むので、登録アドレスの変更を容易に行うことができる。

【0114】ここで、図7及び図8を参照して、検索テーブルの変更（追加、更新、削除を含む。）動作の一例として、32ビットのIPアドレスを8ビットずつ四つの分割ビット列に分割した場合の本発明によるIPアドレスの登録方法について説明する。

【0115】図7及び図8は、IPアドレスの追加登録方法を説明するためのフローチャートである。図7では、登録しようとするIPアドレスのプレフィックス長が8ビット以下の場合である場合に、IPアドレスを登録する手順を示している。この場合には、第二テーブルと一段目第一テーブルとに対してのみ、登録することとなる。すなわち、登録にあたっては、まず、デフォルトの第二テーブルを検索する（図7のS1）。ここでは、第二テーブルの先頭エントリへアクセスする。

【0116】次に、第二テーブルにおいて、IPアドレスに対応する第一テーブルを示す第一テーブルポインタが登録済みか否かを判断する（図7のS21）。ここでは、この先頭エントリに、登録しようとするIPアドレスの先頭ビット値に対応する0ビット又は1ビット用第一テーブルポインタが設定されているか否かを判断する。例えば、登録しようとするIPアドレスが、プレフィックス長「1」の「8. 0. 0. 0」である場合について説明する。この場合、先頭ビット値は「1」であるので、1ビット用第一テーブルポインタ用フィールドの内容を確認する。

【0117】そして、先頭エントリに該当する第一テーブルポインタが設定されている場合には、そのポインタの示す一段目第一テーブルへアクセスする（図7のS22）。例えば、1ビット用第一テーブルポインタ用フィールドに第一テーブルポインタが設定されている場合には、そのポインタの示す1ビット用第一テーブルへアクセスする。

【0118】一方、先頭エントリに該当する第一テーブルポインタが未設定（NUL）の場合には、第二テーブルに、新規の第一テーブルを示す第一テーブルポインタを設定する（図7のS23）。ここでは、第二テーブルの先頭エントリの1ビット用テーブルポインタ用フィールドに、未だ使われていない新規の1ビット用第一テーブルを示す第一テーブルポインタを設定する。そして、新規の第一テーブルポインタの示す一段目第一テーブルへアクセスする（図7のS24）。

【0119】次に、第一テーブルで登録し得る範囲を決定する（図7のS3）。この第一テーブルは、図7のS22のステップで第二テーブルの先頭エントリの第一テーブルポインタに従ってアクセスした第一テーブル、又

は、図7のS24のステップで、第二テーブルの先頭エントリに新たに設定した第一テーブルポインタに従ってアクセスした新たな第一テーブルである。

【0120】そして、例えば、登録しようとするIPアドレスが、プレフィックス長「1」の「8. 0. 0. 0」である場合について説明する。この場合、成就した従来の拡散方の場合と同様に、エントリアドレスのビット列のうち、1ビット目が「1」である上位8ビットの全ての分割ビット列が登録し得る範囲となる。すなわち、

10 「128 (=1000 0000). 0. 0. 0」～
「255 (=1111 1111)」の各エントリが登録し得る範囲となる。

【0121】次に、登録し得る範囲に対して、更新制御を行う（図7のS4）。具体的には、図7のS3のステップにおいて決定した登録し得る範囲内のエントリに、既に登録済みのIPアドレスに対応する第二テーブルポインタが無い場合には、新しいIPアドレスに対応する第二テーブルポインタを、その範囲内の各エントリに設定する。なお、新たに設定する第二テーブルポインタは、第二テーブルの未使用（NUL）であったエントリを示すものを選ぶ。そして、その選択されたエントリに、その登録IPアドレスのホップポインタを設定し、登録を終了する。

【0122】また、その範囲内に、既に登録済みのIPアドレスがある場合には、まず、登録済みのIPアドレスのプレフィックス長と新たに登録しようとするIPアドレスのプレフィックス長とを比較する。そして、登録済みのプレフィックス長が新たなプレフィックス長以上である場合には、登録済みのままとして、登録を終了する。

【0123】一方、これに対して、登録済みのプレフィックス長が新たなプレフィックス長よりも短い場合には、その範囲のエントリの内容を、新しく登録するIPアドレスに対応する第二テーブルポインタに更新する。なお、更新した第二テーブルポインタとしては、第二テーブルの未使用（NUL）であったエントリを示すものを選ぶ。そして、その選択されたエントリに、その登録IPアドレスのホップポインタを設定し、登録を終了する。

40 【0124】また、図8では、登録しようとするIPアドレスのプレフィックス長が32ビット中、8ビットより長く、16ビット以下の場合である場合に、IPアドレスを登録する手順を示している。この場合には、第二テーブル、一段目及び二段目第一テーブルに対して登録することとなる。この場合は、図7のS3のステップに至るまでの処理は、上述した処理と同様である。そして、図7のS3のステップにおいて、以下の処理を行う。

【0125】まず、登録済みの第二テーブルポインタの有無を確認する（図8のS1）。そして、登録済みの第二テーブルポインタがある場合には、そのポインタの示

す第二テーブルのエントリへアクセスする(図8のS2)。さらに、図7の破線Aで囲んだステップの処理を行う(図8のS3)。

【0126】一方、登録済の第二テーブルポインタが無い場合には、新規の第二テーブルポインタを設定する(図8のS4)。続いて、新規に設定された第二テーブルポインタの示す第二テーブルのエントリの内容を設定する(図8のS5)。

【0127】次に、図7のS3のステップと同様にして、第一テーブルで登録し得る範囲を決定する(図8のS6)。続いて、登録し得る範囲に対して、図7のS4のステップと同様にして、更新登録を行い(図8のS7)、登録を終了する。

【0128】なお、登録するIPアドレスのプレフィックス長が16ビットより長く24ビット以下の場合や、24ビットよりも長く32ビット以下の場合も、同様にして、エントリの設定を行うことができる。また、上述した実施形態では、IPアドレスの変更として追加登録の例について説明したが、登録アドレスの更新や削除についても同様に、第一テーブルにおいては、原則、第二

テーブルポインタの変更だけで行うことができる。

【0129】このように、各第一テーブルの該当エントリにおいては、第二テーブルポインタのみを変更すれば良く、ホップポインタなどの転送先に関する情報の変更は、第二テーブルでのみ行えば済む。したがって、本発明によれば、登録アドレスの変更を容易に行うことができる。

【0130】[第二実施形態] 一般に、送信先アドレスに割り当てられたビット列の長さに対して、登録アドレスの数が少ない場合には、一つの第一テーブルに登録されるエントリ数が少なくなる。この場合、一つの第一テーブルあたりの登録エントリ数が一つだけとなる場合が少なからず発生する。このような場合、たった一つの登録エントリのために、分割ビット列分の全エントリを有する第一テーブルのメモリ領域を確保することは、メモリ容量の浪費となる。

【0131】そこで、本発明の第二実施形態では、メモリ容量を節約するために単独第一テーブルを設けた例について説明する。図9は、第二実施形態の転送先検索方法を説明するための検索テーブルの概念構成図である。図9に示すように、第二実施形態では、四段目第一テーブル140が、単独第一テーブルとなっている。

【0132】ここで、図10を参照して、通常の第一テーブルと比較して、単独第一テーブルについて説明する。図10の(A)は、通常の第一テーブルのデータ構成図である。また、図10の(B)は、単独第一テーブルのデータ構成図である。

【0133】図10の(A)に示すように、通常に第一テーブルは、複数のエントリからなり、登録アドレスの分割ビット列値と一致するアドレスのエントリに、その

登録アドレスに対応する第二テーブルポインタがそれぞれ設定されている。図10の(A)に示す例では、エントリアドレスが(000 0001)の1番目のエントリと、エントリアドレスが(111 1110)の126番目のエントリとに、それぞれ第二テーブルポインタが設定されている。

【0134】これに対して、図10の(B)に示すように、単独第一テーブルには、エントリ一つ分だけのメモリ領域の記憶容量を有する。そして、このエントリに、登録アドレスに対応した第二テーブルポインタを設定している。このように、単独第一テーブル140を設ければ、一つの第一テーブルあたりの登録エントリが一つだけの場合に、メモリ容量を節約することができる。なお、単独第一テーブルの場合には、エントリアドレスは、登録アドレスの分割ビット列と一致させる必要はない。

【0135】また、単独第一テーブル140を設けた場合には、その単独第一テーブル140を示す第一テーブルポインタが設定されている、第二テーブルのエントリ(図9の矢印jで指し示されているエントリ)のテーブル種類フラグが、単独第一テーブルである旨を示している。さらに、そのエントリの0ビット用及び1ビット用単独第一テーブルの登録アドレス及びそのプロフィックス長用フィールドには、単独第一テーブル140の単独エントリに登録されている登録アドレスの分割ビット列値とそのプロフィックス長とが設定されている。

【0136】そして、第二実施形態におけるホップポインタ検索処理は、図9の矢印jの処理までは、上述した第一実施形態における処理と同一である。そして、図9の矢印jで示す第二テーブルのエントリにおいて、テーブル種類フラグが単独第一テーブル140であることを示しているのを、以下の処理行う。

【0137】すなわち、図5のS8又はS11のステップのように、第二制御部60(図1参照。)は、そのエントリのフィールドからプレフィックス長を読み出す。そして、その段の分割ビット列値の上位からプレフィックス長分のビット列値部分が、単独第一テーブルに対応する登録アドレスの分割ビット列値と一致するか否かを判断する(図5のS8又はS11)。

【0138】一致した場合、第二制御部60は、第一テーブルポインタを第一制御部50へ転送する(図5のS12)。そして、第一制御部50は、第一テーブルポインタの示す四段目第一テーブル(単独第一テーブル)140へアクセスする(図9の矢印k)。さらに、単独第一テーブル140のエントリの第二テーブルポインタの示す、第二テーブルのエントリへアクセスする(図9の矢印m)。続いて、その第二テーブルのエントリのホップポインタを読み出す(図9の矢印n)。なお、一致しなかった場合、第二制御部60は、検索失敗として、検索を終了する。

【0139】【第三実施形態】次に、図11を参照して、本発明の第三実施形態について説明する。図11は、第三実施形態の転送先検索方法を説明するための検索テーブルの概念構成図である。上述した第二実施形態では、単独第一テーブル140（図9参照。）を設けた例について説明したが、第三実施形態では、その単独第一テーブル140を省略する。そして、第二テーブル20aにおいて、当該単独第一テーブルを示す第一テーブルポインタの代わりに、その単独第一テーブルのエントリに設定されるべき第二テーブルポインタを設定する。

【0140】このようにすれば、単独第一テーブルを省略することができるので、メモリ容量の一層の節約を図ることができる。また、省略された第一テーブルへのアクセス回数分だけアクセス回数を減らすことができるので、検索の迅速化を図ることができる。

【0141】【第四実施形態】次に、図12を参照して、本発明の第四実施形態について説明する。図12は、第四実施形態の転送先検索方法を説明するための検索テーブルの概念構成図である。第四実施形態では、第一テーブルのエントリは、ホップポインタを更新する必要がある場合にのみ、第二テーブルポインタを有し、更新不要の場合には、第二テーブルポインタの代わりに、登録アドレスのビット列値に従って次にアクセスする次段の第一テーブルを示す第一テーブルポインタを有する。

【0142】ここでは、二段目第一テーブル12のエントリが、第二テーブルポインタの代わりに第一テーブルポインタを有し、他段の第一テーブル11、13及び14のエントリは、第二テーブルポインタを有している。そして、ホップポインタを検索するにあたり、二段目第一テーブル12から三段目第一テーブル13へ、第二テーブルを経由せずに、直接アクセスする（図12の矢印q）。

【0143】このように、更新フラグが更新不指示の場合に、第二テーブルポインタの代わりに、次段の第一テーブルを示すテーブルポインタを格納しておけば、第二テーブルにアクセスせずに、次段の第一テーブルへアクセスすることができる。その結果、検索中のアクセス回数を減らすことができるので、検索の高速化を図ることができるとともに、そのテーブルポインタの分だけ、第二テーブルのエントリ数を減らすことができるので、第二テーブルのメモリ容量の節約を図ることもできる。

【0144】上述した実施の形態においては、本発明を特定の条件で構成した例について説明したが、本発明は、種々の変更を行うことができる。例えば、上述した説明ではIPv4（ビット長：32ビット）を例にして話しているが、IPv6（ビット長：128ビット）の場合でも同様な効果が得られる。IPv4及びIPv6混在の場合でも、IPv4のビット列の前に96ビットのビット列（但し、IPv6では使用されていないビット

列）を追加して、IPv4をIPv6の一部部分することに対応可能である。なお、IPv6の場合、送信先アドレスの長さが128ビットなので、8ビットの第一テーブルの場合、最大第16段の構成となる。

【0145】さらに、第二テーブルエントリにIPv4単独、IPv6単独及びIPv4／IPv6共用等のフィールドを追加し、第二テーブルのエントリそのものをIPv4及びIPv6で共用することもできる。その場合、第二テーブルのフィールドにIPv4／IPv6の識別フラグ用フィールドを追加するとよい。そして、識別フラグの値は、IPv4用、IPv6用、IPv4及びIPv6両方用の三つの値をのいずれかとすることが望ましい。

【0146】また、上述した実施形態では、送信先アドレスのビット列を8ビットずつに分割した例について説明したが、この発明では、各分割ビット列のビット数は、8ビットに限定されず、任意好適なビット値とすることができる。また、上述した実施形態では、各段の分割ビット列のビット数を均一としたが、この発明では、段によって分割ビット数が異なっても良い。

【0147】なお、上述した実施形態において図5に示すフローチャートでは、第二テーブルのエントリすべてを読込んでいるが、Bit(X)の値は一段目～四段目の各段の検索において、それぞれ対応する送信先アドレスのビットより判明するので、第二テーブルのエントリを読み出す場合に、必要な部分のみを読み出して実行することも可能である。

【0148】

【発明の効果】以上、詳細に説明したように、本発明によれば、階層化された第一テーブルの他に、インデックスとしての第二テーブルを設け、第一テーブルと第二テーブルとを交互に検索する。そして、次に転送先を示すホップポインタを第二テーブルにのみ格納し、第一テーブルは、第二テーブルポインタのみをエントリ内容とする。

【0149】これにより、個々の第一テーブルを構成するメモリ容量を、従来の拡張法における個々の検索テーブルのメモリ容量よりも少なくして、検索テーブル全体のメモリ容量を節約することができる。

【0150】また、本発明では、インデックスとしての第二テーブルにのみホップポインタを設けている。このため、登録アドレスの変更にあたっては、膨大な第一テーブルの該当エントリでは、第二テーブルポインタのみを変更すれば良く、ホップポインタなどの転送先に関する情報の変更は、第二テーブルでのみ行えば済む。したがって、本発明によれば、登録アドレスの変更を容易に行うことができる。

【図面の簡単な説明】

【図1】第一実施形態の転送先検索装置の構成を説明するための機能ブロック図である。

【図 2】第一実施形態の転送先検索方法を説明するための検索テーブルの概念構成図である。

【図 3】第一テーブルのエントリ例を示すデータ構成図である。

【図 4】第一実施形態における第二テーブルのエントリの構成を示すデータ構成図である。

【図 5】第二テーブルにアクセス時の動作例を説明するためのフローチャートである。

【図 6】ホップポイントの更新の有無を説明するための概念図である。

【図 7】登録送信先アドレスの変更方法を説明するためのフローチャートである。

【図 8】登録送信先アドレスの変更方法を説明するためのフローチャートである。

【図 9】第二実施形態の転送先検索方法を説明するための検索テーブルの概念構成図である。

【図 10】(A)は、通常的第一テーブルのデータ構成図であり、(B)は、単独第一テーブルのデータ構成図である。

【図 11】第三実施形態の転送先検索方法を説明するための検索テーブルの概念構成図である。

【図 12】第四実施形態の転送先検索方法を説明するための検索テーブルの概念構成図である。

【図 13】従来のバイナリ・ツリー検索法を説明するための概念図である。

【図 14】従来のラディックス・ツリー検索法を説明するための概念図である。

【図 15】従来の拡張法を説明するための概念図である。

【図 16】従来の拡張法における更新例を説明するためのフローチャートである。

【図 17】従来の拡張法における更新例を説明するためのフローチャートである。

【符号の説明】

10 第一テーブル群

11、12、13、14、110、140 第一テーブル

20、20a、20b 第二テーブル

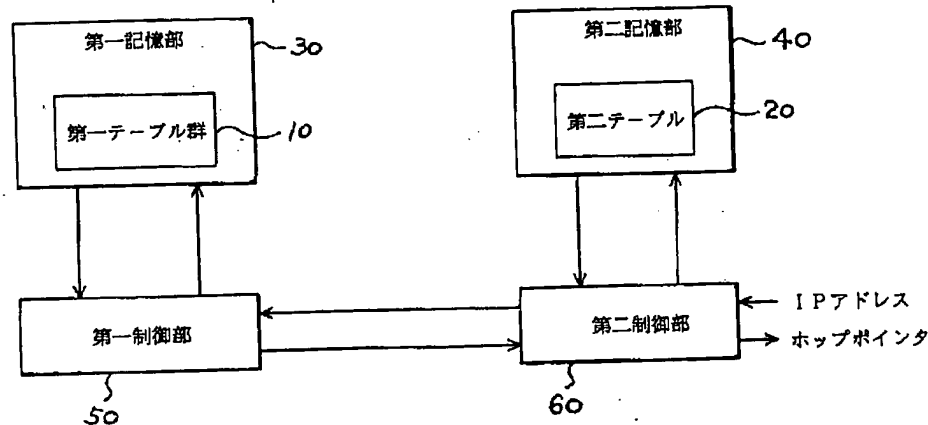
30 第一記憶部

40 第二記憶部

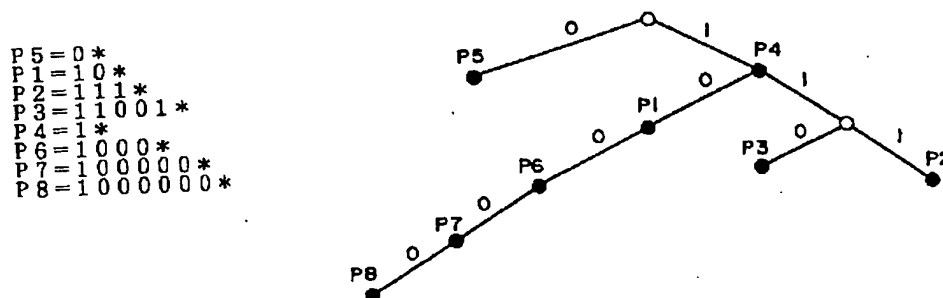
50 第一制御部

60 第二制御部

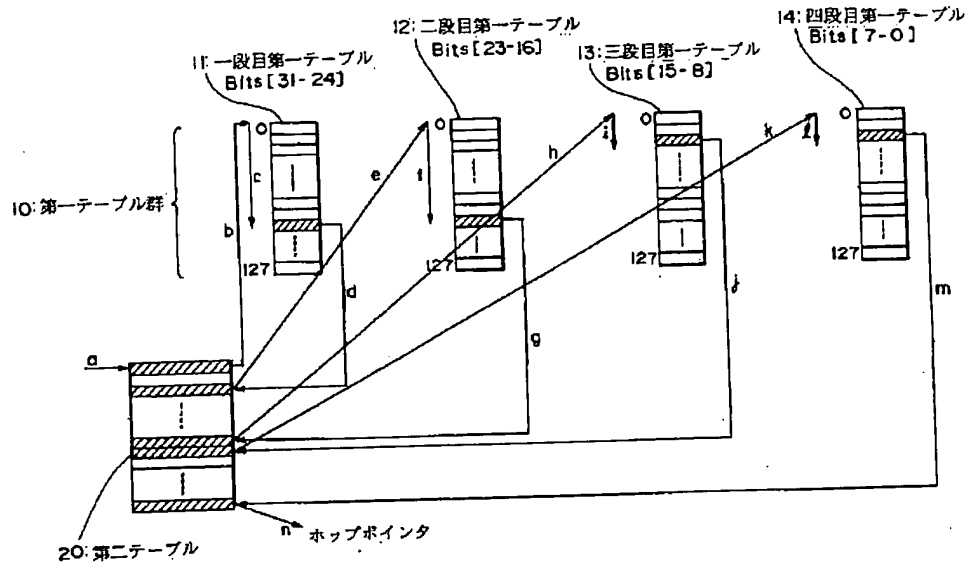
【図 1】



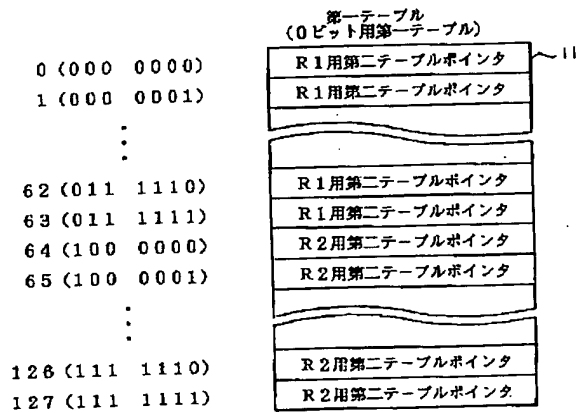
【図 14】



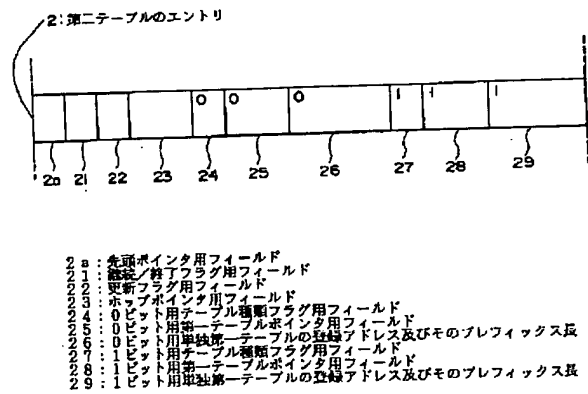
【図 2】



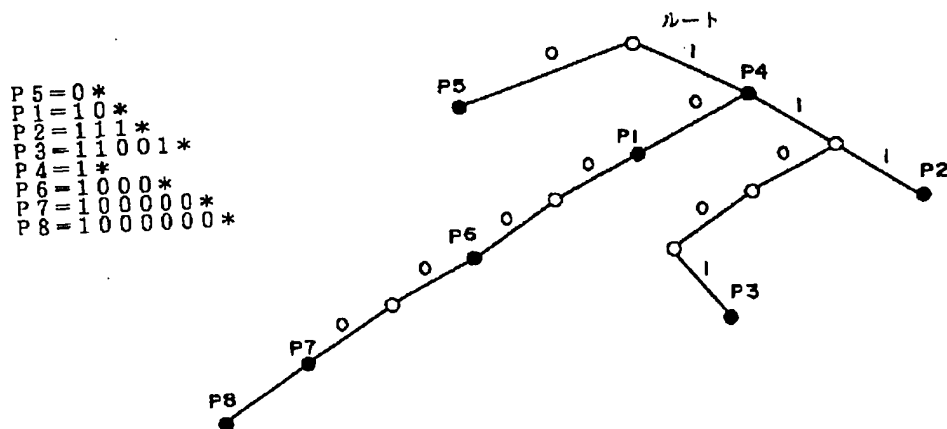
【図 3】



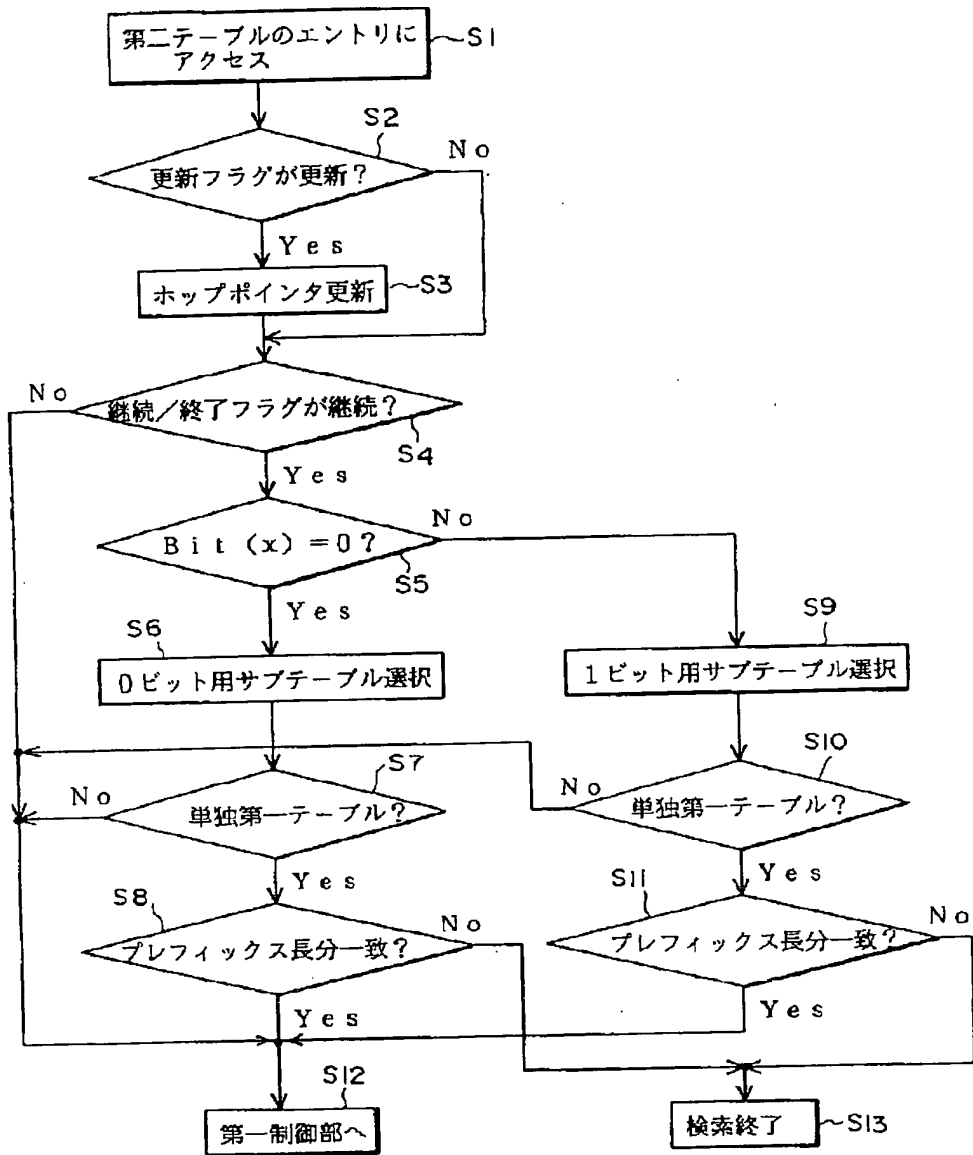
【図 4】



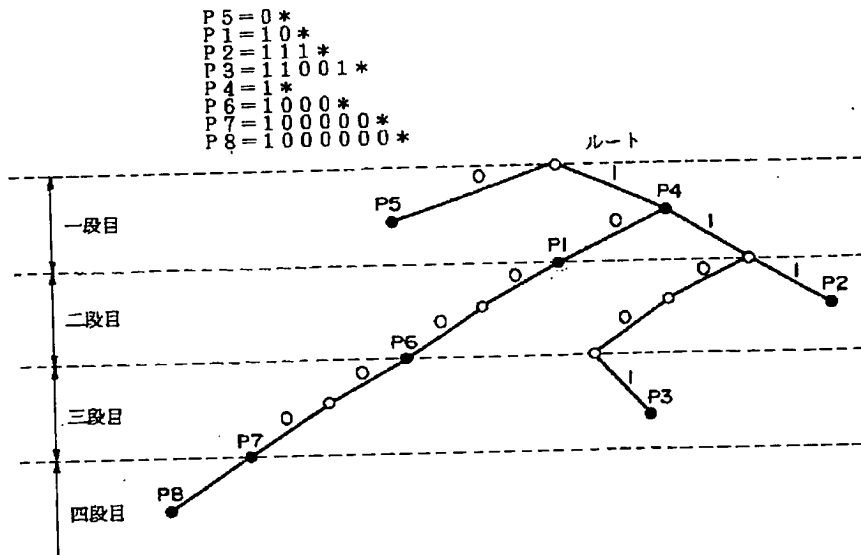
【図 13】



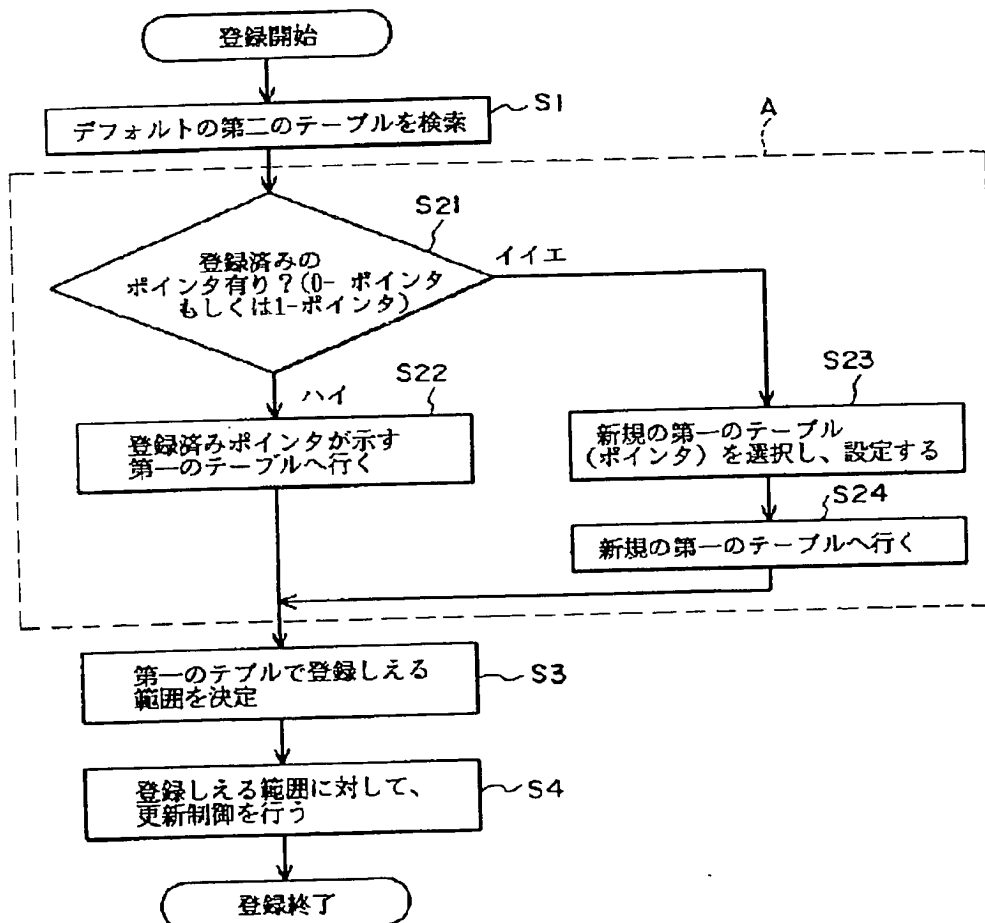
【図5】



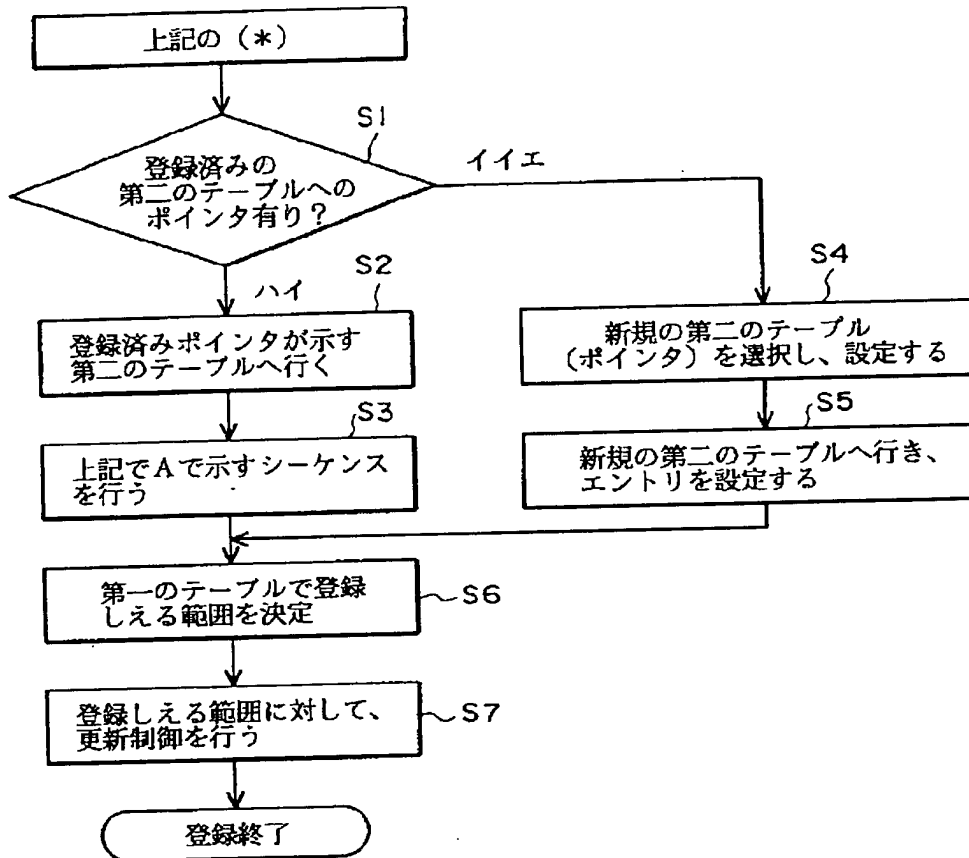
【図 6】



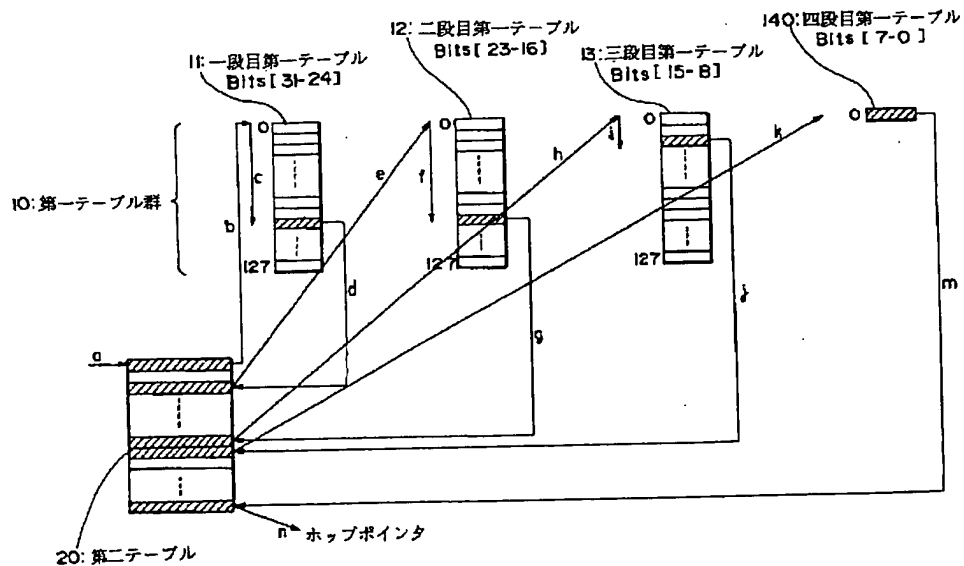
【図 7】



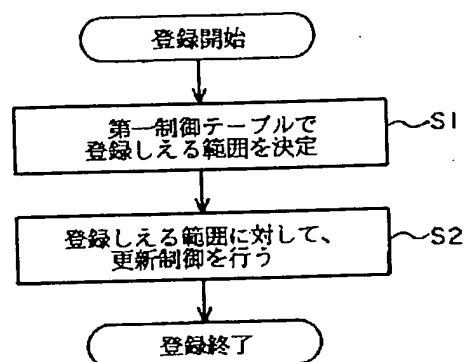
【図 8】



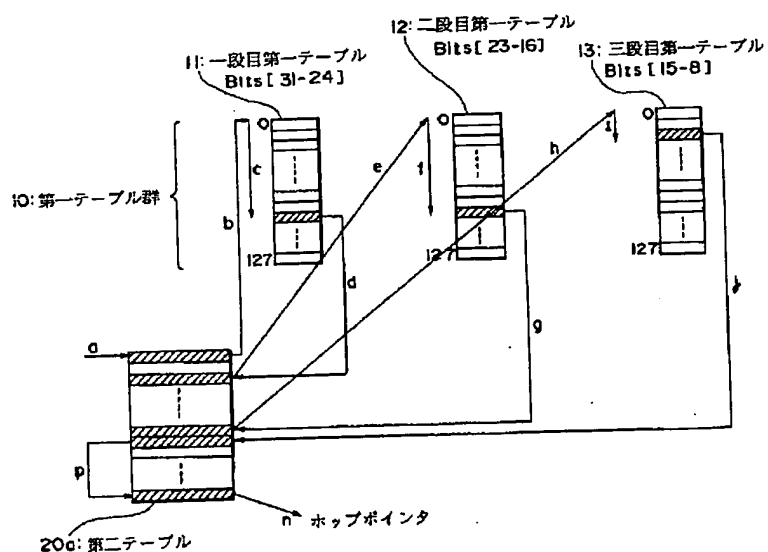
【図 9】



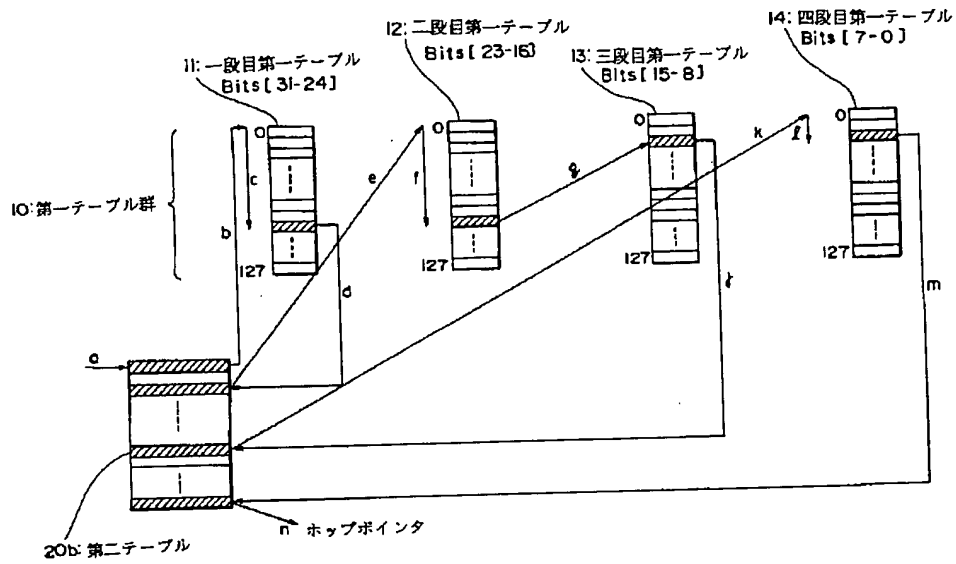
【图 16】



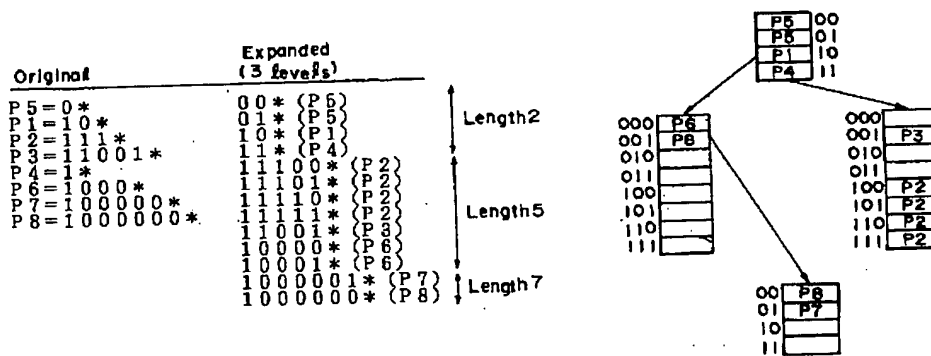
【※ 1 1】



【図 12】



【図 15】



【図17】

